UC Berkeley
UC Berkeley Previously Published Works

Title
Generating Music Reactive Videos by Applying Network Bending to Stable Diffusion

Permalink
https://escholarship.org/uc/item/3g12n0wp

Journal
Journal of the Audio Engineering Society, 73(6)

ISSN
1549-4950

Authors
Dzwonczyk, Luke
Cella, Carmine-Emanuele
Ban, David

Publication Date
2025

DOI
10.17743/jaes.2022.0210

Copyright Information
This work is made available under the terms of a Creative Commons Attribution-ShareAlike License, available at https://creativecommons.org/licenses/by-sa/4.0/

Peer reviewed

eScholarship.org Powered by the California Digital Library
University of California

L. Dzwonczyk, C.-E. Cella, and D. Ban, "Generating Music-Reactive Videos by Applying Network Bending to Stable Diffusion," *J. Audio Eng. Soc.*, vol. XX, no. X, pp. 1–11 (2025 X). https://doi.org/10.17743/jaes.2022.0210.

Generating Music-Reactive Videos by Applying Network Bending to Stable Diffusion

LUKE DZWONCZYK,* CARMINE-EMANUELE CELLA, AND DAVID BAN
(dz.luke@berkeley.edu) (carmine.cella@berkeley.edu) (djban@berkeley.edu)

Center for New Music and Audio Technologies University of California, Berkeley Berkeley, CA

This paper presents the first steps toward the creation of a tool which enables artists to create music visualizations using pretrained, generative, machine learning models. First, the authors investigate the application of network bending, the process of applying transforms within the layers of a generative network, to image generation diffusion models by utilizing a range of point-wise, tensor-wise, and morphological operators. A number of visual effects that result from various operators, including some that are not easily recreated with standard image editing tools, are identified. The authors find that this process allows for continuous, fine-grain control of image generation, which can be helpful for creative applications. Next, music-reactive videos are generated using Stable Diffusion by passing audio features as parameters to network bending operators. Finally, the authors comment on certain transforms that radically shift the image and the possibilities of learning more about the latent space of Stable Diffusion based on these transforms. This paper is an extended version of the paper "Network Bending of Diffusion Models," which appeared in the 27th International Conference on Digital Audio Effects.

0 INTRODUCTION

The present authors seek to create an artistic tool that aids in the creation of music visualizations: videos in which aspects of the image change in relation to aspects of the sound. A system is proposed that generates music-reactive videos given a sound file and some constraints. The system, which utilizes generative diffusion models [1], is flexible enough to create a wide variety of visual aesthetics. It can produce abstract textures and shapes as well as specific objects and scenes and can move between different visual aesthetics within the same video. The authors' hope is that the system creates a relationship between sound and image that is clear but complex. This paper presents preliminary steps toward these goals and investigates an implementation that shows promise while acknowledging that there is still more work to be done to create such a system.

Today, more and more artists work across disciplines and modalities, bridging the gaps between different types of media [2, 3]. Various areas of study and artistic domains have sprung up at these intersections, such as audiovisual art [4, 5]. From the perspective of a composer or musician, it may be desirable to bring other art forms, such as visual art, into one's practice [6, 7]. Music visualizations can complement a piece of music by bringing it into a new modality.

One avenue for a composer to realize a music visualization is by collaborating with a visual artist. For example, the composer and artist Max Cooper, who is known for his music videos, works with a different visual artist for each of his videos. While these collaborations can be extremely fruitful and fulfilling for both sides, there can also be a desire for a single person to create both the sound and the visuals. This may lead to a more unified approach where the artist, working alone, can more fully realize their creative idea.

In the authors' opinion, it is important to note that by shutting themselves off from collaboration, the lone artist will be passing up opportunities to have their view of the piece expanded by working with another artist. It should not be forgotten that collaboration can be an extremely beneficial working method.

Nonetheless, if there is a desire to have more control over the creation of both the audio and visuals, then a composer may find themselves lacking the technical skills to create visual art; it is difficult for a single person to have expertise in both fields. Of course, this is not impossible as some individuals, such as artist Ryoji Ikeda, possess the skills to create both music and visual art. However by allowing one to create both visuals and music, it is possible for the artist

*To whom correspondence should be addressed, email: dz.luke@berkeley.edu

to have conceptual unity across the two modalities. Nothing is lost in translation.

In this proposed system, the artist can seek a specific visual aesthetic that is represented semantically using text or images. This aesthetic can be applied to the system as constraints on the generation of images. In order to apply these semantic constraints, the authors look to machine learning methods.

In the field of music information retrieval (MIR), there has been a shift from using hand-crafted features to using machine-learned features, which has opened up new possibilities in audio representations [8]. In the same way, the authors seek to push music visualization past the phase of hand-crafted one-to-one mappings and into the area of machine learned mappings and semantics. Working in the pixel domain, just like working in the waveform domain, only allows certain operations or effects to be applied to an image or a sound. Just as one cannot remove the sound of one source from a complex auditory scene using standard audio methods, one cannot change the background of an image using standard image editing methods. In order to achieve such results, different methods are necessary; the authors need to work not at the pixel level but at the semantic level. It is this idea that guides this work.

As SEC. 1 will outline, a standard approach to generating music visualizations is to map the audio features of the music to the visual features of the video. In this way, any number of audio features such as amplitude, pitch, noisiness, etc. can control any visual parameter such as color, brightness, etc. While this approach is a valid one, the authors search for a deeper mapping that is not a one-to-one, one-to-many, or many-to-one mapping from audio features to visual features. Their hope is that the complexity of this mapping allows for more meaningful and compelling visualizations.

In SEC. 2, this work begins by utilizing diffusion models for image generation. For a standard text-to-image diffusion model, the only control that a user has over the resulting image is the text prompt and the random seed. The random seed gives no expressive control to the user since there is no continuity; changing the seed by 1 has the same effect as changing it by 1,000. Therefore, the user can only control image generation through the text prompt. While this gives great semantic control, it does not give continuous or fine-grain control of the image. Small changes in the text prompt can lead to large changes in the resulting image. The authors seek a level of continuous control that follows the Lipschitz continuity: a small change in the input should lead to a small change in the output. Since sounds are continuous and often have smooth changes, this is an important aspect of this system.

In order to reach this goal, the authors propose the application of network bending to be applied to pretrained diffusion models as a method of exerting creative control over the output of the model. Network bending, proposed by [9], allows this control by applying transformations within the layers of the network during generation, giving the user the ability to influence output through one or multiple changing parameters.

As a first step, the authors investigate using network bending to affect the generation of images from a text-to-image diffusion model. Various functions are identified that are capable of applying different visual transformations to images, illustrating that network bending can be applied to diffusion models. SEC. 2.1 lists the different operators experimented with, and the visual effects that result from these operators are identified in SEC. 3.

In SEC. 2.2, the authors seek to use network bending to generate audio reactive videos. This is done through frame by frame generation by an image generation model, where the creation of each frame is influenced by the current audio at the time the frame is displayed. The generated frames are then stitched together, and the audio that conditioned the generation plays simultaneously. Short videos are generated that take an audio file and text prompt(s) as input. After choosing an operator to be used for network bending and an audio feature to be passed as a parameter to the operator, music-reactive videos are created. In order to achieve a more complex relationship between audio and image, machine-crafted audio features are experimented with as well.

To summarize, this paper's contributions are as follows:

- It shows for the first time that network bending can be applied to diffusion models in order to exert expressive control over image generation.
- It shows the variety of visual effects that different transformations have on image output.
- It shows that videos can effectively be created using image generation models and that music-reactive visualizations can be created using network bending.

The code is provided at https://github.com/dzluke/JAES2025. A series of videos and supplementary images that were generated can be viewed at https://dzluke.github.io/JAES2025.

1 STATE OF THE ART

A music visualization is the realization of a sonic and time-based phenomenon through light, color, shapes, or symbols. There are many different approaches to music visualization: the use of video and animation, lights, and lasers, created using software or hardware. Visualizations can be static images or dynamic videos. They can be created in real time or precomputed, composed or algorithmically generated. This paper will focus on visualizations that are digital and created using software. Systems that are generative and can be real-time or offline will be identified.

Broadly speaking, visualizations fall into two categories: functional and aesthetic [10]. Common in MIR, the goal of a functional visualization is to provide new information to the viewer, aid in analysis of a sound, or show the sound in a new light [11]. Aesthetic visualization, on the other hand, is concerned with the creation of art. This paper seeks the aesthetic visualization of sound; the goal is to create art.

The line between functional and aesthetic visualizations can be blurred. For example, the spectrogram itself is a type

of music visualization, often thought of as functional but used for artistic means as well [12]. Arc diagrams toe the line between aesthetic and functional, visualizing the form of different musical pieces by connecting repeated sections in a way that reveals something new about the piece in an artistic way [13].

1.1 Classical Methods

Many methods have been employed to create both functional and aesthetic visualizations. Often they involve an analysis of a sound and a representation of that analysis through visual forms. For example, in [14], similarities between different sections in a piece of music can be visualized by calculating the Mel-frequency cepstral coefficients of a segment and computing a similarity measure to all other segments in the piece. The self-similarity matrix that arises out of this is visualized as an image. The authors in [15] apply principal component analysis to audio features and then use self-similarity and self-organizing maps to achieve various visualization methods, some real-time, for the purposes of music classification.

Within the realm of aesthetic visualization, a common approach to creating dynamic music visualizations is for the artist to create a mapping from audio features to visual features [16, 17]. For example, the amplitude and spectral centroid could be mapped to the color and texture, respectively, of some objects on screen. The beats per minute could control a rate of movement of these objects, and, using beat detection, they could move around the screen on the beat. This mapping could be saved as a preset, and different presets could be used for different types of music. This approach to visualization can be used for both real-time and precomputed visualizations.

Common softwares for creating real-time visualizations include Jitter and TouchDesigner, which allow the creation and linking of modules that perform different computational tasks and are fast enough to create images on the fly [18]. Many libraries and plug-ins, such as Vsynth[1] for MaxMSP and Scintillator for SuperCollider, allow artists to create visuals through preset or custom functions and can take input from any number of audio streams or sensor-based sources.

1.2 Learning-Based Methods

Another approach to creating visualizations is through the use of machine learning, which can be used to generate images and videos. Generative adversarial networks (GANs), which consist of a discriminator network and generator network, are able to generate images of a single class [19] and have been employed in various ways to create music visualizations. The authors in [20] train an encoder-decoder model to perform music-to-image and then use the resulting image to apply style to an input image in a style-transfer process. A major limitation of this approach is that the stylization effect does not change over time but determines a single visual style based on the musical input and is therefore not suited for dynamic music visualization.

Another GAN-based approach, TräumerAI [21], uses a convolutional neural network as a music encoder and translates music embeddings into a visual embedding space and then generates images using StyleGAN2 [22]. This approach is effective for creating dynamic visualizations; however, there does not appear to be a strong temporal synchronization between audio and video. In a system that resembles the present goal, the authors of [23] apply network bending and other techniques to StyleGAN2 to create music-reactive videos that change based on various audio features. The major difference between this system and the one proposed here is that the present authors use diffusion models and hope to create a system that does not use hand-picked mappings between audio features and visual characteristics. In a real-time example, the output of an audio synthesis long short-term memory network is encoded in a variational autoencoder and passed as input to a modified GAN that is continuously interpolating between images [24].

While some systems show promise for generalized visualization, using GANs is avoided here for a number of reasons. First, GANs are usually specific to one style or object type, like paintings, puppies, or Van Gogh, instead of a generalized image feature space. The reason for this is that the authors wish the user to be able to move between visual aesthetics within one video, for example, from an impressionist painting to a 3D rendering to a graphite drawing. Standard GANs are also unable to generate images conditioned on text or image, which gives the user less semantic control over the generation.

More recently, diffusion models have been employed for image generation. Diffusion models work by training a network to remove noise from images, and when pure noise is fed to the model, it can be guided by a text prompt to generate an image of that prompt [25]. These models have been used for creating music visualizations in a number of ways. AudioToken [26] is capable of performing audio-to-image, generating an image that reflects the source of a sound, such as a picture of a bird when a bird song is input. MM-Diffusion [27] jointly generates video and audio, for example, generating a video of the ocean and the sounds of waves lapping at the shore. However, both of these examples are functional visualizations, aiming to create an image that provides information about a sound, but the authors seek aesthetic visualizations.

Generative Disco [28] is the closest example to the present goal: a video that moves through different text prompts is generated, and the interpolation speed between text prompts is determined by the amplitude of percussive elements at a given point in time. This system is built from a modified version of Stable Diffusion capable of generating music-reactive videos.[2] The main drawbacks are that the only visual feature that is changing is the interpolation between prompts and the only audio feature being employed

[1] https://www.kevinkripper.com/vsynth

[2] https://github.com/nateraw/stable-diffusion-videos

is the amplitude. The authors seek a system that has a complex relationship between the timbral elements of the audio and the visual characteristics of the image, not a one-to-one mapping.

2 METHODOLOGY

The previous section discussed how diffusion models have shown results for image generation and promise for audio visualization. Therefore, Stable Diffusion, an open-source text-to-image diffusion model, is used to generate all examples shown in this paper [29]. The architecture of Stable Diffusion consists of three distinct networks: a text encoder, diffusing U-Net, and image decoder.

Network Bending is applied in the layers of the U-Net, which is where the diffusion process takes place. At any point in the diffusion process, the image being diffused is represented by a compressed encoding, which is a tensor of shape (4, 64, 64). These tensors are then input to an operator, and the transformed output is fed to the next layer. By applying network bending, parameterized control of the output image is enabled, which is not possible otherwise.

There are four parameters that define an individual application of network bending [9], each of these parameters can have an effect on the resulting image:

1. Layer: the operator can be applied before or after any layer of the network.
2. Operator: can be a point-wise, tensor-wise, or morphological transformation.
3. Parameter: most operators take a parameter as input, such as the scalar to multiply by or the angle to rotate by.
4. Feature: the operator can be applied to all elements of the latent tensor, a single dimension of the tensor, or a random selection of its features.

2.1 Experiments

In order to test the different parameters that can affect network bending, images are first generated. Each image has one transform applied at one layer. Many of the transformations applied are taken from [9]. A grid search is performed on the parameter space of each operator, and parameters that lead to images that are entirely black are disregarded. Unless otherwise noted, the point-wise functions are applied to every element of the latent tensor.

All images are generated using pretrained Stable Diffusion v1[3] with the frozen v1.4 checkpoint[4]; there is no additional training or fine-tuning performed. The denoising diffusion implicit models sampler is tested with the default setting of 50 sampling steps and the seed set to 46.

2.1.1 Point-Wise Operations

The authors apply numerous point-wise operators that transform each element of the latent tensor and select four operators that lead to meaningful visual change in the resulting image. For each function, the input x is one element of the latent tensor, and r is a parameter of the given operator.

1. Addition of a scalar: $f(x) = x + r$
2. Multiplication by a scalar: $f(x) = x \cdot r$
3. Hard threshold: $f(x) = \begin{cases} 1 & \text{if } x \geq r \\ 0 & \text{otherwise} \end{cases}$
4. Inversion: $f(x) = \frac{1}{r} - x$

2.1.2 Tensor Operations

Another type of transformation experimented with are tensor operations, in which an operator tensor is contracted with the latent tensor, applying an operation in the same way a matrix multiplication would. These operations can be thought of as shifting the latent tensor to a new location in the feature space. The two tensor operations applied are rotation and reflection.

Rotation is applied by contracting a 4×4 rotation matrix with the latent tensor. In R_4, there are six possible planes of rotation. R_{ij} is used to denote a rotation in the ith and jth plane. When generating videos with machine-crafted features in SEC. 2.2.1, a matrix is used that rotates in all six planes simultaneously, which is called R_6.

Reflection is applied through four different 4×4 reflection matrices, in which each one reflects across one dimension. A reflection matrix is the 4×4 identity matrix with one of the elements on the diagonal set to −1.

2.1.3 Morphological Transformations

Finally, the authors employ two morphological transforms, erosion and dilation, as found in [9]. These are applied to the latent tensor, treating it as a four-channel image. The transformations are implemented using the Kornia library [30].

Overall, these transformations did not lead to as meaningful results as achieved in [9]; however, it was found that normalizing the tensor after applying the transformation led to more promising results. The normalization is done by subtracting the mean from each element, but applied to specific dimensions. For example, if dimension 1 is normalized, then the mean of each row is subtracted from each element in that row.

2.2 Video Generation

After investigating the visual effects that result from different transformations, the same methods are used to generate music-reactive videos. Audio-reactivity is achieved in the video by applying network bending during the generation of each frame. For each frame, there is approximately 30 ms of audio that plays during that frame. A feature of this audio is calculated, and that feature is passed as the parameter to the network bending function. For example, the RMS of the audio is calculated, and that is passed as the angle of rotation to the network bending function $R_{2,3}$.

The following parameters are necessary to generate a video: an audio file to be visualized, a series of text prompts

[3] https://github.com/CompVis/stable-diffusion
[4] https://huggingface.co/CompVis/stable-diffusion-v-1-4-original

with timestamps, the operator to be used in network bending, the layer to perform the bending at, and the audio feature to be passed as a parameter to the network bending operator. In the generation of a single frame, the input passed to Stable Diffusion is comprised of a text encoding and an initialization noise. The text encoding is calculated as a linear interpolation between the previous and next text prompts based on the current time and the user-provided timestamps. The initialization noise is the result of a batched noise process generated in the following way: First a standard normal distribution is sampled to create a two tensors of noise, A and B. Then, to generate frame i out of total of k frames, the initial noise passed to the model equals $A * \sin\frac{2\pi i}{k} + B * \cos\frac{2\pi i}{k}$ [31].

The authors experiment with different standard audio features including RMS, spectral shape (centroid, spread, skewness, kurtosis), and spectral flux. These features are chosen because they are commonly used in MIR tasks and can represent audio with a single value, which is useful since most of the transformations take only one parameter [32].

2.2.1 Machine-Crafted Audio Features

Machine-crafted audio features were also explored as an alternative to the standard features, as a more complex relationship between sound and image is sought. The authors experiment with using EnCodec, a convolutional audio auto-encoder with residual vector quantization, to create compressed representations of the audio [33]. EnCodec is used in MusicGen [34], a text-to-music model, and is trained on music from the MTG-Jamendo dataset [35]. The output of the encoder is taken in EnCodec, and this representation is passed as input to network bending functions. This encoding compresses audio into a latent representation of 150 values, which the authors perform dimensionality reduction on by applying t-SNE to further reduce the representation to 16 or six dimensions, depending on the setup [36].

With 16 dimensions, the values are reshaped into a tensor of size (4, 4), and this tensor is contracted with the latent representation of size (4, 64, 64) within Stable Diffusion, resulting in a tensor of size (4, 64, 64). In this approach, the audio feature is the output of EnCodec, and the network bending operator is tensor contraction. For a reduction to six dimensions, the network bending operator used is R_6, a six plane rotation, with each element of the encoding being the angle of rotation for one of the six planes.

For example, a video[5] is generated with an excerpt from "Singularity" by John Hopkins as audio input, the text prompt "shapes," and the EnCodec audio feature reduced to 16 dimensions with tensor contraction as the network bending operator. Throughout the 30-s video, there are various shapes that do not move but change color and texture. The audio excerpt can be roughly divided into three sections: In the beginning, there is a percussion loop and a mid-frequency to high-frequency pad. The density of layers decreases in the middle section to feature only the percussion loop and a repeated synthesizer hit. Then in the final section, a strong kick drum enters with other low-frequency sounds. The visual characteristics, especially color, follow these three sections: In the beginning, the shapes appear on top of a white, pixelated background. As the higher pitched pad fades out in the middle section, the background turns a much darker shade. In the final section, when the kick drum enters, the background becomes light again and the shapes pulse with the beat.

2.2.2 Smoothing

Applying smoothing to the audio features before passing them to the network bending function greatly improves the quality of the generated videos. The following three smoothing methods are applied, in this order: median filtering, envelope follower, and noise gate. Smoothing is applied to the time series list of audio features.

For median filtering, the `signal.medfilt` function implemented in the scipy library [37] is used. The noise gate sets all values below a certain threshold to zero. The envelope follower uses the following logic:

$$y[n] = \begin{cases} (1-\alpha)x[n] + \alpha y[n-1], & \text{if } x[n] < y[n-1] \\ x[n], & \text{otherwise} \end{cases}, \tag{1}$$

where x is the input audio feature list and y is the smoothed output.

2.2.3 Stream Diffusion

In order to take advantage of the latest developments in the efficient generation of images, Stream Diffusion, a "real-time" adaptation of Stable Diffusion [38], is used. It is found that this greatly speeds up computation time, decreasing the time it takes to generate an image by a factor of 10.

In these experiments with standard Stable Diffusion, it takes approximately 1 s to generate a single frame on an NVIDIA GeForce RTX 4090, meaning a 1-min video at 20 frames per second takes approximately 20 min to generate. With Stream Diffusion, 10 frames per second can be generated, cutting the generation of the same video down to only 2 min.

3 DISCUSSION

In order to create music-reactive videos, the authors need to identify the visual effect that each transformation has on the image. It is found that various operators are capable of a number of different effects, which are summarized in Table 1. A green color filter can be achieved through adding a scalar [Fig. 2(b)], and a saturation effect is achieved through multiplication by a scalar [Fig. 2(d)]. These are standard visual effects that are achievable with media editing software. However, other transforms lead to results that are more complex and not accessible through standard methods. For example, adding a scalar to only 5% of the features can change only the background color of the image [Fig. 2(c)]

[5]The video can be viewed at https://dzluke.github.io/JAES2025

Table 1. A summary of the effect that the transforms typically have on image generation. Each transform does not necessarily apply the effect for every possible input prompt but for the majority that were tested. The erosion and dilation effects can be viewed at https://dzluke.github.io/JAES2025.

Transform	Effect	Example
Addition	Green color filter	Fig. 2(b)
Multiplication	Saturation	Fig. 2(d)
Threshold	Stained glass	Fig. 2e
Inversion	Variety, including scene change	Figs. 2(f)–2(j), Fig. 6(a)
Rotation, reflection	Color shift; semantic shift	Figs. 3 and 4
Erosion, dilation	Variety, including defocus	See website

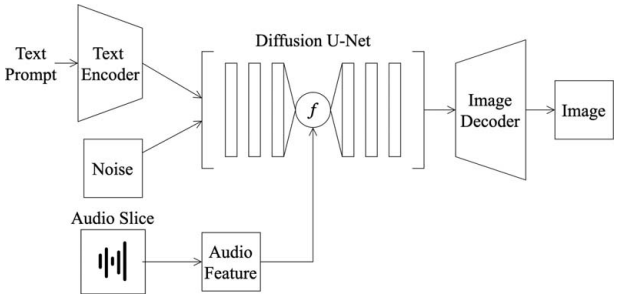

Fig. 1. A block diagram of the system used for video generation. An operator f is inserted at some layer in the U-Net. The audio feature computed from the audio at that time is passed as the parameter to f.

and applying a hard threshold before the last layer creates a stained glass effect [Fig. 2(e)].

The result of applying inversion, as shown in Figs. 2(f)–2(j), leads to a shift in the image which is larger than a filter effect. The authors call this a "scene change": a transformation in which coherency is maintained but a significant shift in the image's contents or style has occurred. As seen in Fig. 2(f), applying inversion before the first layer places the orb in a background of ocean water while changing the size, location, color, and texture of the orb. It is found that multiple transformations are capable of creating scene changes, as can be seen in Fig. 6.

As seen in Fig. 3, applying rotations as tensor-wise operations cycles the image through various color filters. The range of possible colors is determined by the rotation matrix, and the color change is a result of the angle of rotation. Interestingly, as the angle increases from 0 to 2π in $R_{2,3}$, the color filter moves along the color spectrum: yellow, green, blue, indigo, violet, red. While the effect may be a simple color filter, it sometimes does more than this: In Fig. 3(f), a blue filter is applied to the image, and the creation of blue lakes appears in what were previously valleys. The other rotation matrices apply a similar color filter but only between a few colors instead of the full spectrum. For example, $R_{1,3}$ applies an orange or blue filter, depending on the angle. $R_{1,4}$ applies a brown filter and $R_{1,2}$ applies a purple filter. Similar to a rotation, a reflection across one dimension applies a color filter: no change for the first dimension, purple

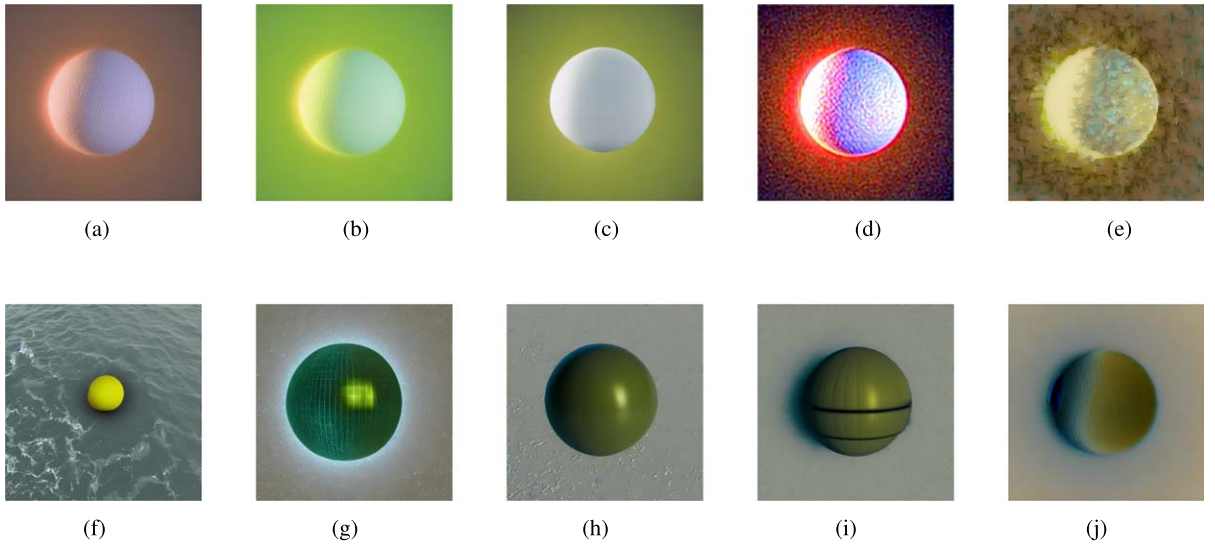

Fig. 2. Image generations using the prompt "a floating orb" with various point-wise operators applied. (a) No operator applied; (b) add scalar, $r = 1$, layer 40; (c) add scalar, $r = 1$, layer 0, random selection of 5% of the features; (d) multiply scalar, $r = 10$, layer 49; (e) hard threshold, $r = 0.1$, layer 49; (f) inversion, $r = 100$, layer 0; (g) inversion, $r = 1{,}000$, layer 3; (h) inversion, $r = 1{,}000$, layer 6; (i) inversion, $r = 1{,}000$, layer 9; (j) inversion, $r = 1{,}000$, layer 49.

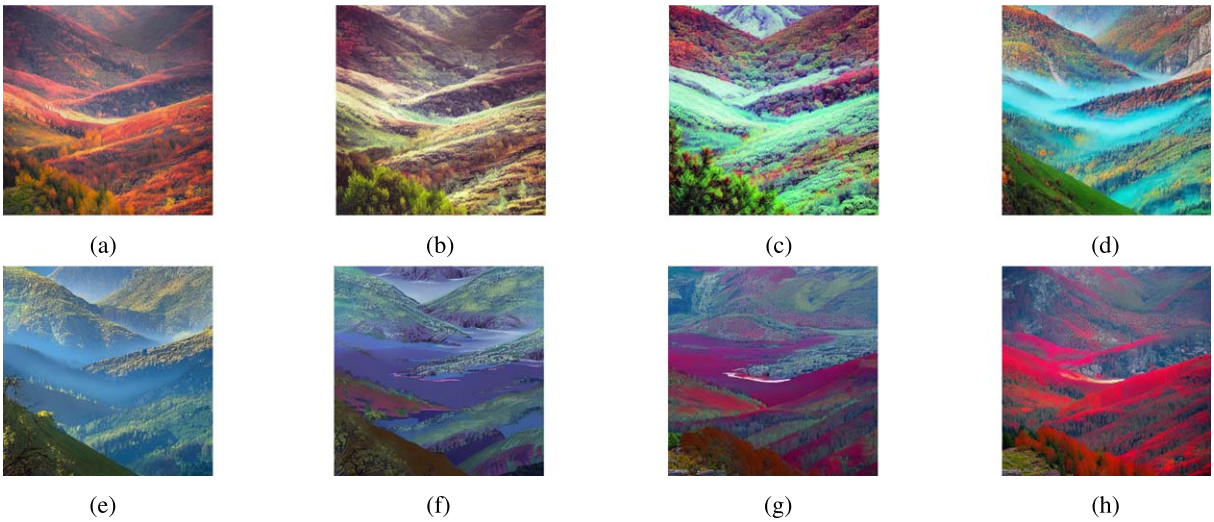

Fig. 3. Image generations using the prompt "a gorgeous landscape" with $R_{2,3}$ applied at layer 20 with changing angle. (a) No operator applied, (b) $\theta = 0.25\pi$, (c) $\theta = 0.5\pi$, (d) $\theta = 0.75\pi$, (e) $\theta = \pi$, (f) $\theta = 1.25\pi$, (g) $\theta = 1.5\pi$, and (h) $\theta = 1.15\pi$.

for the second dimension, blue for the third dimension, and orange for the fourth dimension.

When a rotation is applied to a text prompt that is a homograph, words with the same spelling but different meanings, it is found that the semantic meaning of the image may change. The authors call this effect a "semantic shift." In Fig. 4, this effect can be seen with different text prompts and various angles of rotation. When the prompt "a photo of a crane" is generated with no transformation applied, Stable Diffusion creates an image of a mechanical crane used in construction. If the same prompt is used but a rotation with $R_{2,3}$, $\theta = 0.5\pi$ is applied, an image of a crane bird is generated. Similar results occur with the prompts "bass" and "cricket." For the prompt "a match," there is an image of a matchstick and an abstract image representing some imagined game: four balls and a person holding a bat are seen. A semantic shift also occurs from the reflection and inversion operators. This is consistent because a reflection is similar to a rotation by π and inversion is similar to a reflection across all dimensions.

The morphological operators of erosion and dilation lead to a kaleidoscope-like effect at early layers and a blurring effect at later layers when normalization is applied after the operator. When used with the prompt "a floating orb" at early layers, smaller, multicolored duplicate spheres are created around the central orb.

The authors also experiment with applying transformations to only certain dimensions of the latent vector. When a scalar is added only to the middle row of the tensor, a green bar appears across the middle of the image. This suggests that there is a relationship between the spatial layout of the compressed tensor and the resulting image. Therefore,

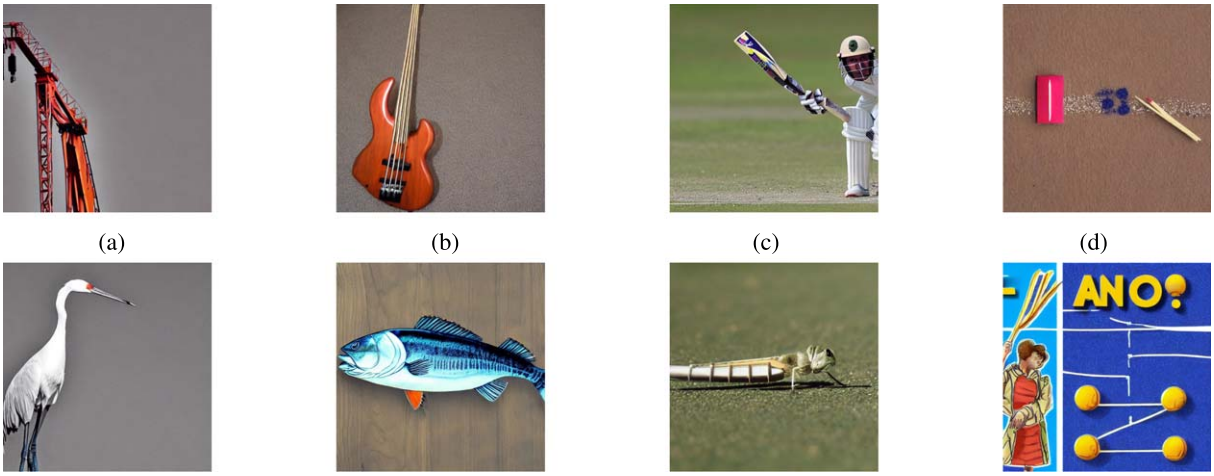

Fig. 4. Examples of semantic shift. The top image is generated with no transformation applied. The bottom image is the same prompt but with $R_{2,3}$ applied at layer 0. (a) "A photo of a crane," (b) "bass," (c) "cricket," (d) "a match," (e) $\theta = 0.5\pi$, (f) $\theta = 0.75\pi$, (g) $\theta = 0.75\pi$, and (h) $\theta = 1.25\pi$.

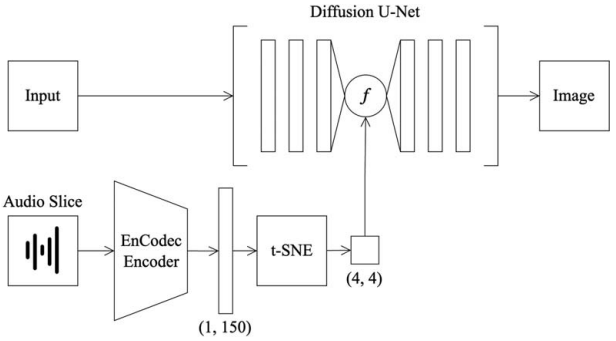

Fig. 5. Block diagram showing the use of EnCodec with t-SNE dimensionality reduction to 16 dimensions, followed by a tensor contraction with the latent representation inside the U-Net.

it may be possible to apply transformations to only a specific part of the image, while leaving the rest of the image untouched.

When generating videos, it is found that using machine-crafted features is a promising path for creating complex relationships between sound and image. For the example video described in SEC. 2.2, the color of the background changes in relation to different sonic layers entering or exiting in the song: it starts with a white background, changes to black in the middle section as the higher-pitched pad fades out, and finally changes back to white when the kick drum enters. At the same time, the shapes in the video pulse with different onsets in the percussion loop. This suggests that the EnCodec features, unlike standard features such as RMS or spectral centroid, may be able to capture musical changes at multiple time scales, which is a complex and important aspect for modeling music.

4 CONCLUSIONS AND FUTURE WORK

The authors propose a tool for the creation of music visualization videos using deep learning. It is found that network bending can successfully be applied to diffusion models and shows promise for allowing continuous, fine-grained control of image generation and the creation of videos. There is a wide range in the complexity of effects that different transformations lead to. Some transformations lead to simple effects such as color filtering or image saturation, yet the authors also achieve transformations that are considerably more advanced: scene changes and semantic shifts. These advanced effects are a strong capability of this system since they are not easily achieved through standard image editing tools. Music-reactive videos are then produced using network bending and passing an audio feature, either hand-crafted or machine-crafted, as a parameter to the network bending operator. It is found that standard, hand-crafted features, such as RMS or spectral flux, are effective for creating these videos and that machine-crafted features, such as the output of an audio auto-encoder, show promise for creating more complex mappings.

Through these experiments, some generalities are found on the effects of different transforms on the latent space. In general, increasing or decreasing the value of the latent tensor leads to changes in color. Increasing, through addition or thresholding, results in an image with more green in it, and decreasing results in the image becoming more purple.

Applying transforms to earlier layers, especially before the first layer, leads to the most dramatic change in the resulting image. This is due to the fact that at later layers in the diffusion process, the image has been mostly formed. At early layers, there is still a potential for a significant shift in the image, since it is still predominantly noise. The scene changes and semantic shifts that are seen occur only if the transform is applied at the earliest layers. Applying certain transformations at the last layer can be useful to apply a specific visual effect while keeping the coherency of the original image.

While experimenting with different transformations, it is found that some tensor operations can lead to a "semantic shift" when the text input is a homograph. This may suggest that concepts that are linked by the same word are laid out in the latent space in a relationship that can be accessed through geometric manipulations. The possibility of a geometry of information [39] in the latent space of Stable Diffusion is extremely preliminary but is an interesting byproduct of this work and may be a path forward for gaining more understanding of the latent space of Stable Diffusion. The authors find it interesting that one of the most simple effects, color filter, and one of the most complex, semantic shift, are a result of the same operation.

There are a number of areas for continued work. First, the authors would like to further explore the area of machine-crafted features, as they believe this shows promise in creating compelling visualizations. This could entail exper-

(a) (b) (c)

Fig. 6. Examples of scene change with various prompts. For each pair of images, the image on the left has no transformation applied. (a) Prompt: "a floating orb," inversion, $r = 30$, layer 0; (b) prompt: "a gorgeous landscape," add scalar, $r = 2$, layer 0, applied to 5% of features; (c) prompt: "mice," rotation by $R_{2,3}$, $\theta = \pi$, layer 0.

imenting with different audio auto-encoders, such as the auto-encoder used in Stable Audio Open, a text-to-audio model [40].

Furthermore, the authors would like to investigate the semantic shift that results from certain transforms in order to better understand the latent space of Stable Diffusion. The invariances of an operator may define a topology defined by the orbit of the operator. This could allow one to create connections between disconnected images through the chaining of operators.

It is difficult to apply quantitative measurements to properly assess the artistic output of this system. However, a possible avenue to explore this could be the use of video distance metrics [41, 42] or assessing the system qualitatively through user studies.

The inclusion of Stream Diffusion, a "real-time" image generator, creates the possibility of generating music-reactive videos in real time, which would open up the possibility of using these methods in a live musical performance.

Finally, there is potential for applying network bending to different types of generative networks, including other image networks, which may give different results based on differences in the latent space. This methodology could also be applied to a video generation network or music generation network [43], which could allow the user to have fine-grain control of the music, perhaps changing timbral, pitch-based, or rhythmic aspects of the music in a continuous way. The authors believe this could be a powerful tool for creative control of text-to-music models.

5 REFERENCES

[1] L. Yang, Z. Zhang, Y. Song, et al., "Diffusion Models: A Comprehensive Survey of Methods and Applications," *ACM Comput. Surv.*, vol. 56, no. 4, paper 105 (2023 Nov.). https://doi.org/10.1145/3626235.

[2] W. Condee, "The Interdisciplinary Turn in the Arts and Humanities," *Issues Interdiscip. Stud.*, vol. 34, pp. 12–29 (2016).

[3] T. Augsburg, "Interdisciplinary Arts," in R. Frodeman (Ed.), *The Oxford Handbook of Interdisciplinary*, pp. 131–143 (Oxford University Press, Oxford, UK, 2017). https://doi.org/10.1093/oxfordhb/9780198733522.013.12.

[4] I. V. Krupskyy, N. I. Zykun, A. P. Ovchynnikova, S. I. Gorevalov, and O. A. Mitchuk, "Determinants and Modern Genres of Audio-Visual Art," *J. Balk. Tribol. Assoc.*, vol. 27, no. 4, pp. 619–636 (2021).

[5] E. Edmonds, A. Martin, and S. Pauletto, "Audio-Visual Interfaces in Digital Art," in *Proceedings of the ACM SIGCHI International Conference on Advances in Computer Entertainment Technology (ACE)*, pp. 331–336 (Singapore) (2004 Jun.). https://doi.org/10.1145/1067343.1067392.

[6] D. Garro, "From Sonic Art to Visual Music: Divergences, Convergences, Intersections," *Organ. Sound*, vol. 17, no. 2, pp. 103–113 (2012 Aug.). https://doi.org/10.1017/S1355771812000027.

[7] J. Watkins, "Composing Visual Music: Visual Music Practice at the Intersection of Technology, Audio-Visual Rhythms and Human Traces," *Body Space Technol.*, vol. 17, no. 1, pp. 51–75 (2018 Apr.). https://doi.org/10.16995/bst.296.

[8] E. J. Humphrey, J. P. Bello, and Y. LeCun, "Feature Learning and Deep Architectures: New Firections for Music Informatics," *J. Intell. Inf. Syst.*, vol. 41, no. 3, pp. 461–481 (2013 Dec.). https://doi.org/10.1007/s10844-013-0248-5.

[9] T. Broad, F. F. Leymarie, and M. Grierson, "Network Bending: Expressive Manipulation of Generative Models in Multiple Domains," *Entropy*, vol. 24, no. 1, paper 28 (2022 Jan.). https://doi.org/10.3390/e24010028.

[10] S. Panda and S. T. Roy, "A Preliminary Model for the Design of Music Visualizations," *arXiv preprint arXiv:2104.04922* (2021 Apr.). https://doi.org/10.48550/arXiv.2104.04922.

[11] H. B. Lima, C. G. R. D. Santos, and B. S. Meiguins, "A Survey of Music Visualization Techniques," *ACM Comput. Surv.*, vol. 54, no. 7 , paper 143 (2021 Jul.). https://doi.org/10.1145/3461835.

[12] D. Monacchi, "Fragments of Extinction: Acoustic Biodiversity of Primary Rainforest Ecosystems," *Leonardo Music J.*, vol. 23, pp. 23–25 (2013 Dec.). https://doi.org/10.1162/LMJ_a_00148.

[13] M. Wattenberg, "Arc Diagrams: Visualizing Structure in Strings," in *Proceedings of the IEEE Symposium on Information Visualization (INFOVIS)*, pp. 110–116 (Boston, MA) (2002 Oct.). https://doi.org/10.1109/INFVIS.2002.1173155.

[14] J. Foote, "Visualizing Music and Audio Using Self-Similarity," in *Proceedings of the 7th ACM International Conference on Multimedia (Part 1)*, pp. 77–80 (Orlando, FL) (1999 Oct./Nov.). https://doi.org/10.1145/319463.319472.

[15] M. Cooper, J. Foote, E. Pampalk, and G. Tzanetakis, "Visualization in Audio-Based Music Information Retrieval," *Comput. Music J.*, vol. 30, no. 2, pp. 42–62 (2006 Jun.). https://doi.org/10.1162/comj.2006.30.2.42.

[16] M. N. Bain, *Real Time Music Visualization: A Study in the Visual Extension of Music*, Master's thesis, Ohio State University, Columbus, OH (2008).

[17] M. F. G. Vieira, *Interactive Music Visualization- Implementation, Realization and Evaluation*, Master's thesis, Universidade da Madeira, Funchal, Portugal (2012 Sep.).

[18] V. J. Manzo, *Max/MSP/Jitter for Music: A Practical Guide to Developing Interactive Music Systems for Education and More* (Oxford University Press, Oxford, UK, 2011). https://doi.org/10.1093/oso/9780199777679.001.0001.

[19] I. Goodfellow, J. Pouget-Abadie, M. Mirza, et al., "Generative Adversarial Nets," *Commun. ACM*, vol. 63, no. 11, pp. 139–144 (2020 Nov.). https://doi.org/10.1145/3422622.

[20] C.-C. Lee, W.-Y. Lin, Y.-T. Shih, P.-Y. P. Kuo, and L. Su, "Crossing You in Style: Cross-Modal Style Transfer From Music to Visual Arts," in *Proceed-*

ings of the 28th ACM International Conference on Multimedia, pp. 3219–3227 (Seattle, WA) (2020 Oct.). https://doi.org/10.1145/3394171.3413624.

[21] D. Jeong, S. Doh, and T. Kwon, "TräumerAI: Dreaming Music With StyleGAN," *arXiv preprint arXiv:2102.04680* (2021 Feb.). https://doi.org/10.48550/arXiv.2102.04680.

[22] T. Karras, S. Laine, M. Aittala, et al., "Analyzing and Improving the Image Quality of StyleGAN," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8107–8116 (Seattle, WA) (2020 Jun.). https://doi.org/10.1109/CVPR42600.2020.00813.

[23] H. Brouwer, "Audio-Reactive Latent Interpolations With StyleGAN," presented at the *4th Workshop on Machine Learning for Creativity and Design at NeurIPS 2020* (Vancouver, Canada) (2020 Dec.).

[24] N. Collins, V. Ruzicka, and M. Grierson, "Remixing AIs: Mind Swaps, Hybrainity, and Splicing Musical Models," presented at the *1st Joint Conference on AI Music Creativity* (Online) (2020 Oct.). https://doi.org/10.30746/978-91-519-5560-5.

[25] J. Ho, A. Jain, and P. Abbeel, "Denoising Diffusion Probabilistic Models," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, pp. 6840–6851 (Vancouver, Canada) (2020 Dec.).

[26] G. Yariv, I. Gat, L. Wolf, Y. Adi, and I. Schwartz, "AudioToken: Adaptation of Text-Conditioned Diffusion Models for Audio-to-Image Generation," *arXiv preprint arXiv:2305.13050* (2023 May). https://doi.org/10.48550/arXiv.2305.13050.

[27] L. Ruan, Y. Ma, H. Yang, et al., "MM-Diffusion: Learning Multi-Modal Diffusion Models for Joint Audio and Video Generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10219–10228 (Vancouver, Canada) (2023 Jun.). https://doi.org/10.1109/CVPR52729.2023.00985.

[28] V. Liu, T. Long, J. Ma, et al., "Digital 'Double Hatters': Augmenting Audiovisual Creative Work With a Generative Text-to-Video Workflow," in *Proceedings of the 58th Hawaii International Conference on System Sciences*, pp. 2664–2677 (Waikoloa Village, HI) (2025 Jan.). https://doi.org/10.24251/HICSS.2025.323.

[29] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-Resolution Image Synthesis With Latent Diffusion Models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10674–10685 (New Orleans, LA) (2022 Jun.). https://doi.org/10.1109/CVPR52688.2022.01042.

[30] E. Riba, D. Mishkin, D. Ponsa, E. Rublee, and G. Bradski, "Kornia: An Open Source Differentiable Computer Vision Library for PyTorch," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 3663–3672 (Snowmass, CO) (2020 Mar.). https://doi.org/10.1109/WACV45572.2020.9093363.

[31] I. Stenbit, "A Walk Through Latent Space With Stable Diffusion," https://keras.io/examples/generative/random_walks_with_stable_diffusion/ (2022 Sep.).

[32] G. Peeters, B. L. Giordano, P. Susini, N. Misdariis, and S. McAdams, "The Timbre Toolbox: Extracting Audio Descriptors From Musical Signals," *J. Acoust. Soc. Am.*, vol. 130, no. 5, pp. 2902–2916 (2011 Nov.). https://doi.org/10.1121/1.3642604.

[33] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High Fidelity Neural Audio Compression," *Trans. Mach. Learn. Res.*, https://openreview.net/forum?id=ivCd8z8zR2 (2023 Sep.).

[34] J. Copet, F. Kreuk, I. Gat, et al., "Simple and Controllable Music Generation," in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pp. 47704–47720 (New Orleans, LA) (2023 Dec.).

[35] D. Bogdanov, M. Won, P. Tovstogan, A. Porter, and X. Serra, "The MTG-Jamendo Dataset for Automatic Music Tagging," presented at the *Machine Learning for Music Discovery Workshop at the 36th International Conference on Machine Learning (ICML)* (Long Beach, CA) (2019 Jun.).

[36] L. van der Maaten and G. Hinton, "Visualizing Data Using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 86, pp. 2579–2605 (2008 Nov.).

[37] P. Virtanen, R. Gommers, T. E. Oliphant, et al., "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nat. Methods*, vol. 17, pp. 261–272 (2020 Mar.). https://doi.org/10.1038/s41592-019-0686-2.

[38] A. Kodaira, C. Xu, T. Hazama, et al., "StreamDiffusion: A Pipeline-Level Solution for Real-Time Interactive Generation," *arXiv preprint arXiv:2312.12491* (2023 Dec.). https://doi.org/10.48550/arXiv.2312.12491.

[39] A. Cont, S. Dubnov, and G. Assayag, "On the Information Geometry of Audio Streams With Applications to Similarity Computing," *IEEE Trans. Audio Speech Lang. Process.*, vol. 19, no. 4, pp. 837–846 (2011 May). https://doi.org/10.1109/TASL.2010.2066266.

[40] Z. Evans, J. D. Parker, C. Carr, et al., "Stable Audio Open," *arXiv preprint arXiv:2407.14358* (2024 Jul.). https://doi.org/10.48550/arXiv.2407.14358.

[41] T. Unterthiner, S. van Steenkiste, K. Kurach, et al., "FVD: A Nnew Metric for Video Generation," presented at the *DeepGenStruct Workshop at at the 36th International Conference on Machine Learning (ICML)* (Long Beach, CA) (2019 Jun.).

[42] H. Martinez, M. C. Farias, and A. Hines, "NAViDAd: A No-Reference Audio-Visual Quality Metric Based on a Deep Autoencoder," in *Proceedings of the 27th European Signal Processing Conference (EUSIPCO)*, pp. 1–5 (A Coruna, Spain) (2019 Sep.). https://doi.org/10.23919/EUSIPCO.2019.8902975.

[43] Z. Evans, C. Carr, J. Taylor, S. H. Hawley, and J. Pons, "Fast Timing-Conditioned Latent Audio Diffusion," in *Proceedings of the 41st International Conference on Machine Learning (ICML)*, pp. 12652–12665 (Vienna, Austria) (2024 Jul.).

THE AUTHORS

Luke Dzwonczyk

Carmine-Emanuele Cella

David Ban

Luke Dzwonczyk is a composer and creative technologist whose work is centered on engaging emerging technologies with music creation and performance. He is currently a Ph.D. student in music at the University of California, Berkeley, where he works closely with the Center for New Music and Audio Technology (CNMAT). His research focuses on the intersection of computational creativity, generative machine learning, and audiovisual art.

•

Carmine-Emanuele Cella is an internationally acclaimed composer with advanced studies in mathematics. For many years, he has worked on the poetical relationships between the structured world of mathematics and the chaotic world of artistic expression, using music as a medium. He studied piano, computer music, composition, mathematics, and philosophy, getting a Ph.D. in composition at the Accademia S. Cecilia in Rome and a Ph.D. in applied mathematics at Università di Bologna. In 2007, Carmine-Emanuele Cella started a collaboration with IRCAM in Paris, where he has been both researcher and composer for many years. Since January 2019, Carmine has been a professor in music and technology at the University of California, Berkeley, where he is also lead researcher at the Center for New Music and Audio Technology (CNMAT). In 2022, he was the recipient of the prestigious Traiettorie music prize for composition.

•

David Ban recently graduated with a B.A. from the University of California, Berkeley in Computer Science and is currently obtaining his master's at Georgia Tech. He previously presented at the International Conference on Digital Audio Effects (DAFx), and he is interested in how artificial intelligence can help humans interact with music in novel ways.