NETWORK BENDING OF DIFFUSION MODELS FOR AUDIO-VISUAL GENERATION

Luke Dzwonczyk and Carmine Emanuele Cella and David Ban,

Center for New Music and Audio Technologies
University California, Berkeley
Berkeley, CA, USA
dz.luke@berkeley.edu

ABSTRACT

In this paper we present the first steps towards the creation of a tool which enables artists to create music visualizations using pre-trained, generative, machine learning models. First, we 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. We identify a number of visual effects that result from various operators, including some that are not easily recreated with standard image editing tools. We find that this process allows for continuous, fine-grain control of image generation which can be helpful for creative applications. Next, we generate music-reactive videos using Stable Diffusion by passing audio features as parameters to network bending operators. Finally, we comment on certain transforms which radically shift the image and the possibilities of learning more about the latent space of Stable Diffusion based on these transforms.

1. INTRODUCTION

We seek to create an artistic tool which aids in the creation of music visualizations: videos in which aspects of the image change in relation to aspects of the sound. We propose a system 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. Our hope is that the system creates a relationship between sound and image that is clear but complex. In this paper, we present preliminary steps towards these goals and investigate 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 audio-visual 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 our 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 have conceptual unity across the two modalities. Nothing is lost in translation.

In our proposed system, the artist can seek a specific visual aesthetic which 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, we 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, we 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; we need to work not at the pixel level but at the semantic level. It is this idea that guides our work.

As we will outline in Section 2, 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, we 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. Our hope is that the complexity of this mapping allows for more meaningful and compelling visualizations.

Copyright: © 2024 Luke Dzwonczyk et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, adaptation, and reproduction in any medium, provided the original author and source are credited.

In Section 3, we begin this work 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. We 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. For example, a visual effect such as saturation can be applied to an image, with a parameter giving fine-grain, continuous control. Since sounds are continuous and often have smooth changes, this is an important aspect of our system.

In order to reach this goal, we propose the application of network bending to be applied to pre-trained 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, we investigate using network bending to affect the generation of images from a text-to-image diffusion model. We identify various functions that are capable of applying different visual transformations to images, illustrating that network bending can be applied to diffusion models. In Section 3.1, we list the different operators we experiment with and identify the visual effects that result from these operators in Section 4.

In Section 3.2, we seek to use network bending to generate an audio reactive video. 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. We generate short videos that take an audio file and text prompt 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, we create music-reactive videos. In the future, the operator and audio feature would not be hand-picked but machine-crafted, as we detail in Section 5.

To summarize, our contributions are as follows:

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

We provide our code at `https://github.com/dzluke/DAFX2024`. A series of videos and supplementary images that we generated can be viewed at `https://dzluke.github.io/DAFX2024/`.

2. 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 pre-computed, composed or algorithmically generated. In this paper, we will focus on visualizations which are digital and created using software. We will identify systems that are generative and can be real-time or offline.

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. In this paper, we seek the aesthetic visualization of sound; our 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]. Martin Wattenberg's "The Shape of Song" toes 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[1].

2.1. 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 [13] similarities between different sections in a piece of music can be visualized by calculating the MFCCs 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. In [14] the authors apply PCA 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 [15, 16]. For example, the amplitude and spectral centroid could be mapped to the color and texture, respectively, of some objects on screen. The BPM 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 presents could be used for different types of music. This approach to visualization can be used for both real-time and pre-computed 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 [17]. Many libraries and plug-ins, such as Vsynth[2] for MaxMSP and Scintillator[3] 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.

2.2. Learning-based Methods

Another approach to creating visualizations is through the use of machine learning, which can be used to generate images and videos.

[1] `https://www.turbulence.org/Works/song/`
[2] `https://www.kevinkripper.com/vsynth`
[3] `https://scintillatorsynth.org/`

Generative Adversarial Networks (GANs), which consist of a discriminator network and a generator network, are able to generate images of a single class [18] and have been employed in various ways to create music visualizations. In [19], the authors 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 [20], uses a CNN as a music encoder and translates music embeddings into a visual embedding space, and then generates images using StyleGAN2 [21]. 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 our goal, the authors of [22] 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 we propose is that we use diffusion models and hope to create a system that does not use hand-picked mappings between audio features and visual characteristics.

While some systems show promise for generalized visualization, we avoid using GANs 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 we 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 [23]. These models have been used for creating music visualizations in a number of ways. AudioToken [24] 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 [25] 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 we seek aesthetic visualizations.

Generative Disco [26] is the closest example to our 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[4]. The main drawbacks are that the only visual feature that is changing is the interpolation between prompts and the only audio feature being employed is the amplitude. We 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.

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

3. METHODOLOGY

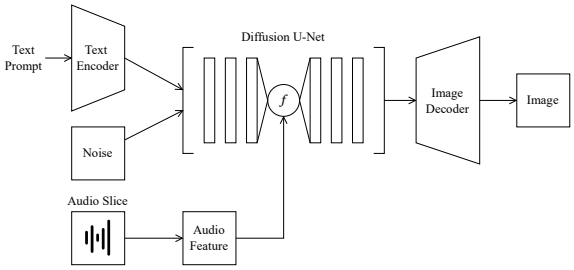


Figure 1: A block diagram of our system. An operator f is inserted at some layer in the U-Net. The audio feature computed on the audio at that time is passed as the parameter to f.

In the previous section, we saw how diffusion models have shown results for image generation and promise for audio visualization. Therefore, we use Stable Diffusion, an open-source text-to-image diffusion model, to generate all examples shown in this paper [27]. Stable Diffusion can generate images in multiple ways; the methods relevant to us are text-to-image and image-to-image. The architecture of Stable Diffusion consists of three distinct networks: a text encoder, a diffusing U-Net, and an 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, we enable parameterized control of the output image, which is not possible otherwise.

There are four parameters that define an individual application of network bending:

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

Each of these parameters can have an effect on the resulting image [9].

3.1. Experiments

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

All images are generated using pre-trained Stable Diffusion v1[5] with the frozen v1.4 checkpoint[6]; we do not perform any additional training or fine-tuning. We use the DDIM sampler with the default setting of 50 sampling steps and the seed set to 46. In our experiments it takes approximately 1 second to generate a single frame on an NVIDIA GeForce RTX 4090, meaning a one minute video at 20 FPS takes approximately 20 minutes to generate.

3.1.1. Point-wise Operations

We apply numerous point-wise operators which 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: $f(x) = \frac{1}{r} - x$

3.1.2. Tensor Operations

Another type of transformation we experiment with are tensor operations, in which an operator tensor is contracted with the latent tensor, applying an operation in the same way as 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 we apply are rotation and reflection.

Rotation is applied by contracting a rotation matrix with the latent tensor. We experiment with the following 4x4 matrices:

$$R_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad R_2 = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_3 = \begin{bmatrix} \cos\theta & 0 & 0 & \sin\theta \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\sin\theta & 0 & 0 & \cos\theta \end{bmatrix} \quad R_4 = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Reflection is applied through four different 4x4 reflection matrices, in which each one reflects across one dimension. A reflection matrix is the 4x4 identity matrix with one of the elements on the diagonal set to -1.

3.1.3. Morphological Transformations

Finally, we employ two morphological transforms, erosion and dilation, as found in [9]. These are applied to the latent tensor, treating it as a 4-channel image. The transformations are implemented using the Kornia library [28].

Overall these transformations did not lead to as meaningful results as achieved in [9], however we 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.

[5] https://github.com/CompVis/stable-diffusion

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

3.2. Audio-to-Video

After investigating the visual effects that result from different transformations, we use Stable Diffusion to generate videos in two distinct ways: using text-to-image with batched noise and using image-to-image with the previous frame as input.

The first method for creating videos uses standard text-to-image generation but changes the initial noise that is input to the system [29]. The initial noise is generated in the following way: first a standard normal distribution is sampled to create a two tensors of noise, which we call A and B. Then, to generate frame i out of total of k frames, the initial noise passed to the model equals $A * \sin\frac{2\pi i}{k} + B * \cos\frac{2\pi i}{k}$. When generating videos with text-to-image, the user supplies a single text prompt or two text prompts to interpolate between. If two prompts are given, the video will start at the first prompt and end at the second prompt. This image interpolation is achieved through linearly interpolating between the text encodings of the two prompts.

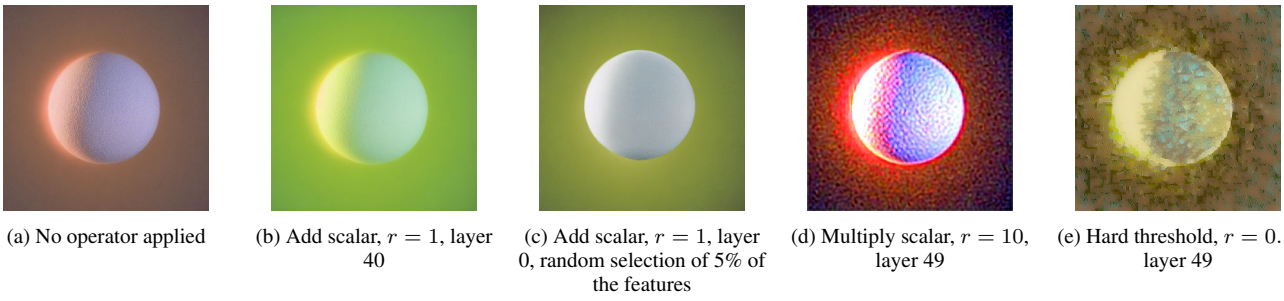
The second method uses image-to-image to create videos. Image-to-image is a process in which the input to the diffusion U-Net is a text prompt, an initialization image, and a "strength" parameter. The diffusion process starts from the initialization image, which has had noise added to it [27]. The amount of noise added is in relation to the strength parameter: a value of 0 corresponds to no noise being added to the image and a value of 1 means the initialization image will be turned into complete noise and have no effect on the resulting image.

To generate videos using image-to-image, the user must provide either an initialization image or a text prompt. The first frame of the video is either the initialization image or the image generated from passing the text prompt to text-to-image. The generation of each subsequent frame is conditioned on the previous frame, using image-to-image, and with an empty string as the text prompt.

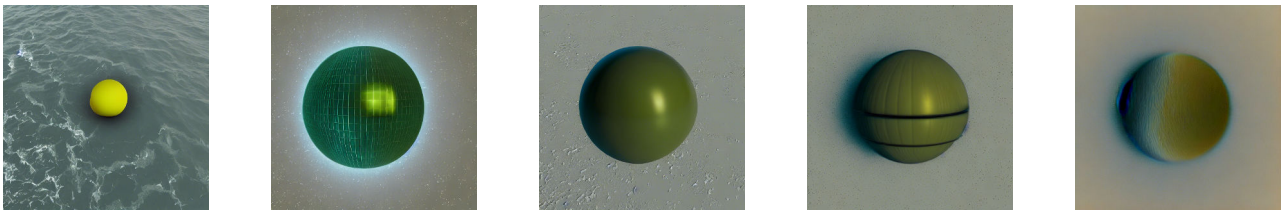
For both methods, we achieve audio-reactivity in the video by applying network bending during the generation of each frame with a user defined operator and audio feature. For a given frame, there is a 50 millisecond window of audio that will play while that frame is shown. The chosen audio feature is calculated for this specific window of audio and is then passed as a parameter to the chosen operator. Usually, the value must be scaled to a different range, as the range of values that give meaningful results for a given operator is not necessarily the same range of the audio feature. We experiment with different audio features including RMS, spectral shape (centroid, spread, skewness, kurtosis), and spectral flux. We choose these features because they are commonly used in MIR tasks and can represent audio with a single value, which is useful since our transformations take only one parameter [30].

For example, we generate a video[7] using text-to-image with solo piano as audio input, the prompt "3D mesh geometry," and apply the rotation R_1 at layer 40 with the RMS of the audio being passed as the angle of rotation. The RMS is scaled to the range 0 to 2π before being passed to the operator. On each note onset from the piano, the image responds through shifting colors which emerge from the black and white mesh. These visual changes follow the amplitude envelope of the piano, with a strong shift in color at the attack and a decay to the original image with the piano. When the piano is quiet or silent, the black and white mesh continues to change as a result of the initialization noise that is fed to the generation.

[7] The video can be viewed at https://dzluke.github.io/DAFX2024/

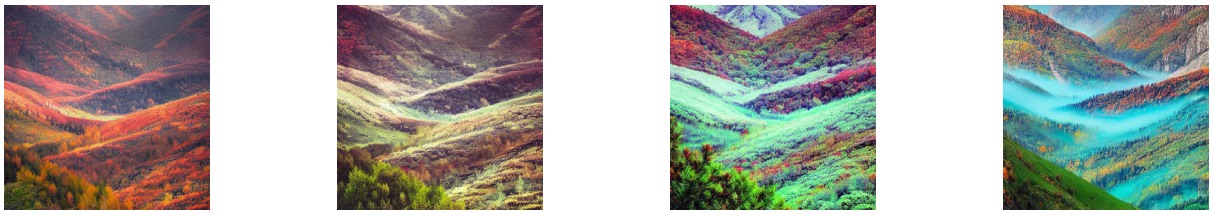


(a) No operator applied (b) Add scalar, $r = 1$, layer 40 (c) Add scalar, $r = 1$, layer 0, random selection of 5% of the features (d) Multiply scalar, $r = 10$, layer 49 (e) Hard threshold, $r = 0.1$, layer 49

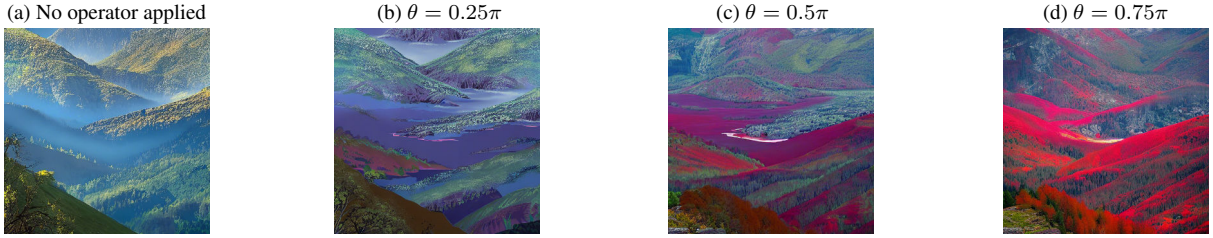


(f) Inversion, $r = 1000$, layer 0 (g) Inversion, $r = 1000$, layer 3 (h) Inversion, $r = 1000$, layer 6 (i) Inversion, $r = 1000$, layer 9 (j) Inversion, $r = 1000$, layer 49

Figure 2: Image generations using the prompt "a floating orb" with various point-wise operators applied



(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) $\theta = 1.5\pi$ (h) $\theta = 1.75\pi$

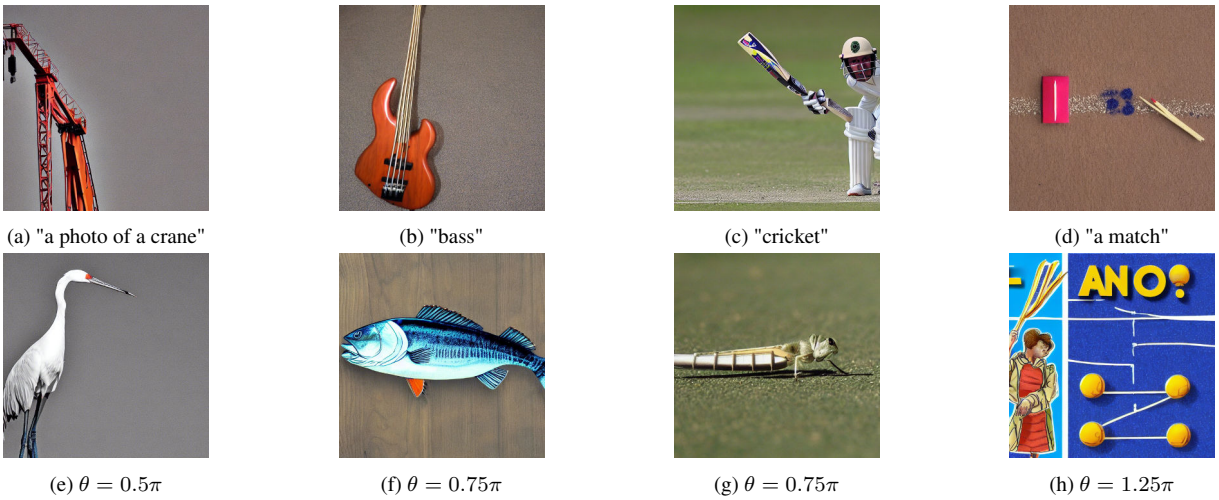
Figure 3: Image generations using the prompt "a gorgeous landscape" with R_1 applied at layer 20 with changing angle

4. DISCUSSION

In order to create music-reactive videos, we need to identify the visual effect that each transformation has on the image. We find that various operators are capable of a number of different effects. A green color filter can be achieved through adding a scalar (Figure 2b), and a saturation effect is achieved through multiplication by a scalar (Figure 2d). 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 (Figure 2c) and applying a hard threshold before the last layer creates a stained glass effect (Figure 2e).

The result of applying inversion, as shown in Figures 2f - 2j, leads to a shift in the image which is larger than a filter effect. We 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 we see in Figure 2f, 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. We find that multiple transformations are capable of creating scene changes, as can be seen in Figure 5.

As seen in Figure 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_1, 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 Figure 3f, a blue filter is applied to the image and the creation of blue lakes appears in what were previously valleys. The



(a) "a photo of a crane" (b) "bass" (c) "cricket" (d) "a match"

(e) $\theta = 0.5\pi$ (f) $\theta = 0.75\pi$ (g) $\theta = 0.75\pi$ (h) $\theta = 1.25\pi$

Figure 4: Examples of semantic shift. The top image is generated with no transformation applied. The bottom image is the same prompt but with R_1 applied at layer 0

other rotation matrices apply a similar color filter, but only between a few colors instead of the full spectrum. For example, R_2 applies an orange or blue filter, depending on the angle. R_3 applies a brown filter and R_4 applies a purple filter. Similar to a rotation, a reflection across one dimension applies a color filter: no change for the first dimension, purple for the second dimension, blue for the third dimension, and orange for the fourth dimension[8].

When a rotation is applied to a text prompt that is a homograph, words with the same spelling but different meanings, we find that the semantic meaning of the image may change. We call this effect a "semantic shift." In Figure 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 $R_1, \theta = 0.5\pi$ is applied, an image of a crane bird is generated. Similar results occur with the prompts "bass" and "cricket." For the prompt "a match," we have an image of a matchstick and an abstract image representing some imagined game: we see four balls and a person holding a bat. 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, multi-colored duplicate spheres are created around the central orb.

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

[8]See examples at `https://dzluke.github.io/DAFX2024/`

5. CONCLUSIONS AND FUTURE WORK

We propose a tool for the creation of music visualization videos using deep learning. We find 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 we also achieve transformations that are considerably more advanced: scene changes and semantic shifts. These effects can be produced through different operators, including inversion, rotation, reflection, and adding a scalar. These advanced effects are a strong capability of our system since they are not easily achieved through standard image editing tools.

Through our experiments, we find some generalities 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 we see 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.

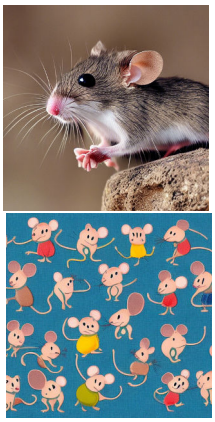
While experimenting with different transformations, we find that some tensor operations can lead to a "semantic shift" when the text input is a homograph. This may suggest that concepts which are linked by the same word are laid out in the latent space in a relationship that can be accessed through geometric manipulations. The possibility of a geometry of information [31] in the latent space of Stable Diffusion is extremely preliminary but is an interesting byproduct of our work and may be a path forward for



(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_1, $\theta = \pi$, layer 0

Figure 5: Examples of scene change with various prompts. The upper image has no transformation applied.

gaining more understanding on 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.

As we noted earlier, the work shown in this paper are the first steps towards the realization of a system for music visualization. An important next step is to employ machine-crafted operators instead of hand-picked transforms. One possible approach to this is to feed the audio into an auto-encoder which outputs a compressed encoding, which is then applied as an operator on the latent tensor. We would also like the semantic constraints applied by the user to be a collection of text, images, or videos. These constraints could define a subspace of the latent space that is navigated during image generation. Similarly, the user could provide specific time points at which each prompt is displayed, and our system could interpolate between these prompts, allowing for temporal and narrative control of the video.

Furthermore, we 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 our system. However, we would like to explore this further through employing video distance metrics [32, 33] and also assess our system qualitatively through user studies. It may be possible to improve the quality of our videos using image upscaling techniques, applying smoothing to the audio features, and allowing for multiple text prompts in the same video [34].

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. Our methodology could also be applied to a video generation network or music generation network [35], 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. We believe this could be a powerful tool for creative control of text-to-music models.

6. ACKNOWLEDGMENTS

Special thanks to Xiaojian Sun, Halie Sung, and Juan Lucas Umali.

7. REFERENCES

[1] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang, "Diffusion models: A comprehensive survey of methods and applications," 2024.

[2] William Condee, "The interdisciplinary turn in the arts and humanities.," *Issues in interdisciplinary studies*, vol. 34, pp. 12–29, 2016.

[3] Tanya Augsburg, *Interdisciplinary Arts*, pp. 131–143, 01 2017.

[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.," *Journal of the Balkan Tribological Association*, vol. 27, no. 4, pp. 619 – 636, 2021.

[5] Ernest Edmonds, Andrew Martin, and Sandra Pauletto, "Audio-visual interfaces in digital art," in *Proceedings of the 2004 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, New York, NY, USA, 2004, ACE '04, p. 331–336, Association for Computing Machinery.

[6] Diego Garro, "From sonic art to visual music: Divergences, convergences, intersections," *Organised Sound*, vol. 17, no. 2, pp. 103–113, 2012.

[7] Julie Watkins, "Composing visual music: Visual music practice at the intersection of technology, audio-visual rhythms and human traces," *Body, Space & Technology*, vol. 17, no. 1, pp. 51, Apr. 2018.

[8] Eric J. Humphrey, Juan P. Bello, and Yann LeCun, "Feature learning and deep architectures: new directions for music informatics," *Journal of Intelligent Information Systems*, vol. 41, no. 3, pp. 461–481, 2013.

[9] Terence Broad, Frederic Fol Leymarie, and Mick Grierson, "Network bending: Expressive manipulation of generative models in multiple domains," *Entropy*, vol. 24, no. 1, 2022.

[10] Swaroop Panda and Shatarupa Thakurta Roy, "A preliminary model for the design of music visualizations," *CoRR*, vol. abs/2104.04922, 2021.

[11] Hugo B. Lima, Carlos G. R. Dos Santos, and Bianchi S. Meiguins, "A survey of music visualization techniques," *ACM Comput. Surv.*, vol. 54, no. 7, jul 2021.

[12] David Monacchi, "Fragments of Extinction: Acoustic Biodiversity of Primary Rainforest Ecosystems," *Leonardo Music Journal*, vol. 23, pp. 23–25, 12 2013.

[13] Jonathan Foote, "Visualizing music and audio using self-similarity," in *Proceedings of the Seventh ACM International Conference on Multimedia (Part 1)*, New York, NY, USA, 1999, MULTIMEDIA '99, p. 77–80, Association for Computing Machinery.

[14] Matthew Cooper, Jonathan Foote, Elias Pampalk, and George Tzanetakis, "Visualization in audio-based music information retrieval," *Computer Music Journal*, vol. 30, no. 2, pp. 42–62, 2006.

[15] Matthew N. Bain, "Real time music visualization: A study in the visual extension of music," M.S. thesis, Ohio State University, 2008.

[16] Marco Filipe Ganança Vieira, "Interactive music visualization- implementation, realization and evaluation," M.S. thesis, Universidade da Madeira (Portugal), 2012, AAI28727326.

[17] V J Manzo, *Max/MSP/Jitter for Music: A Practical Guide to Developing Interactive Music Systems for Education and More*, Oxford University Press, 12 2011.

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, Eds. 2014, vol. 27, Curran Associates, Inc.

[19] Cheng-Che Lee, Wan-Yi Lin, Yen-Ting Shih, Pei-Yi (Patricia) Kuo, and Li Su, "Crossing you in style: Cross-modal style transfer from music to visual arts," in *Proceedings of the 28th ACM International Conference on Multimedia*. Oct. 2020, MM '20, ACM.

[20] Dasaem Jeong, Seungheon Doh, and Taegyun Kwon, "Träumerai: Dreaming music with stylegan," 2021.

[21] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila, "Analyzing and improving the image quality of StyleGAN," in *Proc. CVPR*, 2020.

[22] Hans Brouwer, "Audio-reactive latent interpolations with stylegan," in *Proceedings of the 4th Workshop on Machine Learning for Creativity and Design at NeurIPS 2020*, December 2020.

[23] Jonathan Ho, Ajay Jain, and Pieter Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, Eds. 2020, vol. 33, pp. 6840–6851, Curran Associates, Inc.

[24] Guy Yariv, Itai Gat, Lior Wolf, Yossi Adi, and Idan Schwartz, "Audiotoken: Adaptation of text-conditioned diffusion models for audio-to-image generation," *arXiv preprint arXiv:2305.13050*, 2023.

[25] Ludan Ruan, Yiyang Ma, Huan Yang, Huiguo He, Bei Liu, Jianlong Fu, Nicholas Jing Yuan, Qin Jin, and Baining Guo, "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)*, June 2023, pp. 10219–10228.

[26] Vivian Liu, Tao Long, Nathan Raw, and Lydia Chilton, "Generative disco: Text-to-video generation for music visualization," 2023.

[27] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 10684–10695.

[28] E. Riba, D. Mishkin, D. Ponsa, E. Rublee, and G. Bradski, "Kornia: an open source differentiable computer vision library for pytorch," in *Winter Conference on Applications of Computer Vision*, 2020.

[29] Ian Stenbit, "A walk through latent space with stable diffusion," `https://keras.io/examples/generative/random_walks_with_stable_diffusion/`, 2022, Accessed: 2024-03-18.

[30] Geoffroy Peeters, Bruno L Giordano, Patrick Susini, Nicolas Misdariis, and Stephen McAdams, "The timbre toolbox: Extracting audio descriptors from musical signals," *The Journal of the Acoustical Society of America*, vol. 130, no. 5, pp. 2902–2916, 2011.

[31] Arshia Cont, Shlomo Dubnov, and Gérard Assayag, "On the information geometry of audio streams with applications to similarity computing," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 837–846, 2011.

[32] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly, "Fvd: A new metric for video generation," in *DGS@ICLR*, 2019.

[33] Helard Martinez, Mylène C.Q. Farias, and Andrew Hines, "Navidad: A no-reference audio-visual quality metric based on a deep autoencoder," in *2019 27th European Signal Processing Conference (EUSIPCO)*, 2019, pp. 1–5.

[34] Jinseok Kim and Tae-Kyun Kim, "Arbitrary-scale image generation and upsampling using latent diffusion model and implicit neural decoder," 2024.

[35] Zach Evans, CJ Carr, Josiah Taylor, Scott H. Hawley, and Jordi Pons, "Fast timing-conditioned latent audio diffusion," 2024.

[36] Perry R. Cook and George Tzanetakis, "Audio information retrieval (air) tools," in *International Society for Music Information Retrieval Conference*, 2000.

[37] David Bau, Steven Liu, Tongzhou Wang, Jun-Yan Zhu, and Antonio Torralba, "Rewriting a deep generative model," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.