AIMC 2023

YouTube Mirror: An Interactive Audiovisual Installation based on Cross-Modal Generative Modeling

Sihwa Park

Published on: Aug 29, 2023

URL: https://aimc2023.pubpub.org/pub/m5cbcx8i

Abstract

YouTube Mirror is an interactive, audiovisual AI installation that generates images and sounds in response to the images of the audience captured through a camera in real time in the form of an interactive mirror. YouTube Mirror uses a cross-modal generative machine model that was trained in an unsupervised way to learn the association between images and sounds obtained from the YouTube videos I watched. Since the machine represents the world only based on the learned audio-visual relationship, the audience can see themselves through the machine-generated images and sounds. YouTube Mirror is an artistic attempt to simulate my unconscious, implicit understanding of audio-visual relationships that can be found in the videos. As an empirical case study, this project also aims to investigate the possibility of the use of cross-modal generative modeling in the production of interactive audiovisual installations. This paper describes the background of the project and the design, implementation, and exhibition of the YouTube Mirror installation, followed by the discussion and future work.

Figure 1. A YouTube Mirror installation

Introduction

When we watch videos, we try to understand the relationships between images and sounds in the videos. Along with the popularity and impact of video-based social media platforms such as YouTube, we watch a plethora of videos, and our video consumption affects how we perceive and understand the world around us. What videos we watch are not only determined by our choices but also hugely affected by the recommendation algorithms

of the platforms that are intended to make their users remain longer on their platforms. For example, the "watch-to-next" videos suggested by the algorithms are, in general, based on the user's previous watch history and other metadata related to the videos. Since these data could be implicitly biased or wrongly reflect the user's behavior or preference, the recommendation algorithms could cause a feedback loop effect, narrowing down the choices of videos the user can find [1][2]. This feedback loop will affect our understanding of audio-visual relationships that we unconsciously find in videos we watch. As an artistic motivation, this project attempts to represent the world through a machine that learns these audio-visual associations with cross-modal generative modeling.

Cross-Modal Generative Modeling in Audiovisual Art

Generative modeling in machine learning (ML) is an unsupervised learning technique to train a model with an unlabeled dataset and try to learn hidden patterns, called latent representation in the training dataset. Once a generative model learns the probabilistic distribution of a training dataset, the model can generate new data that looks similar to but does not exist in the training dataset by sampling from the model. Variational Autoencoders (VAEs) [3] and Generative Adversarial Networks (GANs) [4] are representative deep learning architectures for generative modeling.

While the remarkable progress in ML for the arts has focused on unimodal generation or text-to-image generation with large language models, such as DALL-E [5], Midjourney[1], and Stable Diffusion [6], audiovisual art, where the transformation and association between sound and images frequently happen, is a less explored realm in ML research. Several noteworthy attempts have been made to generate sound from images and images from sound with generative machine models. Synesthetic Variational Autoencoder (SynVAE) [7] attempts to transform paintings into sounds. Jeong et al. [8] suggest a neural network architecture to generate videos that respond to music by using StyleGAN2 [9]. In both attempts, however, the image-music pairs for training a model are arbitrarily made. Adapting Neural Visual Style Transfer [10] to the audio domain, Odlen et al. [11] and Verma et al. [12] propose a generative model that learns a sound-image relationship from pairings of music and corresponding album cover art. Akten's *Ultrachunk* (2018) [13] shows a cross-modal modeling approach to combining a singer's facial feature changes obtained while the singer was singing songs with her singing voice data.

Compared to the previous attempts explained above, this project investigates the possibility of cross-modal generative modeling based on a more concrete sound-image relationship that can be found in video data, in creating interactive audiovisual art installations.

YouTube Mirror

YouTube Mirror is an interactive, audiovisual AI installation that generates images and sounds in response to the images of the audience captured through a camera in real time, in the form of an interactive mirror.

YouTube Mirror uses a cross-modal generative machine model that was trained in an unsupervised way to learn the association between images and sounds obtained from the YouTube videos I watched.

YouTube Mirror employs a concept of interactive mirror in digital media arts, as a way to represent how machine vision perceives the world. Since the machine represents the world solely based on its learned audio-visual relationship, the audience can see themselves through the lens of the machine's vision with the machine-generated images and sounds.

YouTube Mirror is an artistic attempt to simulate my unconscious, implicit understanding of audio-visual relationships that can be found in and limited by the videos. YouTube Mirror also attempts to represent the possibility of the machine's bias inherent in a small-size video dataset which is possibly affected by video recommendation algorithms on YouTube, in an abstract and artistic way.

Video Data

YouTube Watch History

The video data of YouTube Mirror is based on the YouTube watch history of my Google account, which can be obtained by using the Google Takeout[2] service. The watch history has the metadata of 14,315 videos that I watched from November 11th, 2018 to March 24th, 2022, including title, subtitle, URL, timestamp, etc. The watch history data is in JSON format as shown in Table 1. A substring following "`v=`" in a video URL represents a video ID. For example, the substring `pIjt_z4JHGM` is the ID of the video in Table 1.

```
{
  {
    'activityControls': ['YouTube watch history'],
    'header': 'YouTube',
    'products': ['YouTube'],
    'subtitles': [{'name': 'NBC News',
      'url': 'https://www.youtube.com/channel/UCeY0bbntWzzVIaj2z3QigXg'}],
    'time': '2022-03-24T04:56:20.343Z',
    'title': 'Watched Nightly News Full Broadcast - March 23',
    'titleUrl': 'https://www.youtube.com/watch?v=pIjt_z4JHGM'
  }
}
```

Table 1. An example of the YouTube watch history data

By using the YouTube Data API[3] along with extracted IDs of all the videos, it was found that there are only 12,183 videos that are available and non-redundant among the watched video.

Video Data Collection and Preprocessing

Based on Dale's Cone of Experience [14], it is widely believed that "learners can generally remember 50 percent of what they see and hear." Although this argument was not scientifically justified, this project considers the argument as an artistic concept to choose the amount of data to be used for training a model. Thus, I decided to use 6,091 videos, which is 50 percent of 12,183 videos. One second segment of each video

was randomly extracted and stored by using *youtube-dl*[4], a command-line program to download videos from YouTube.

Design

Cross-Modal Audiovisual Generative Modeling

As an integral part of this project, YouTube Mirror needs to train a machine to learn the audio-visual relationships from the videos. To this end, this project utilizes cross-modal generative modeling.

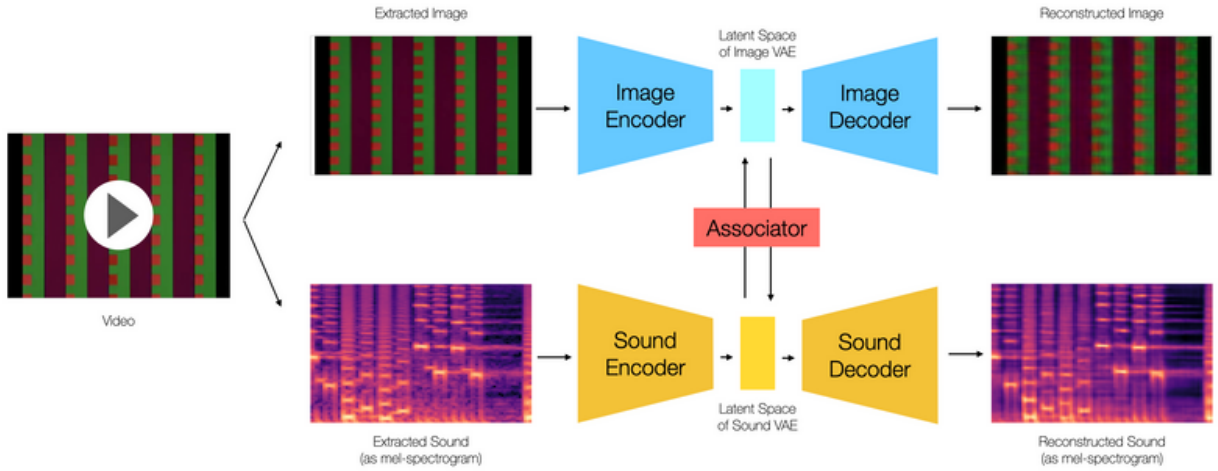

Figure 2: A diagram of cross-modal VAEs with associators

Figure 2 illustrates the model architecture of YouTube Mirror, cross-modal Variational Autoencoders (VAEs) with associators. This architecture is based on an approach proposed by Jo et al. [15]. To briefly explain, VAEs are probabilistic, generative autoencoders. An autoencoder consists of an encoder, a decoder, and the same input and output data with no labels. The encoder reduces high-dimensional input data into low-dimensional latent variables, called latent vectors. The decoder converts latent variables back into high-dimensional space. In this regard, autoencoders are used for dimensionality reduction. While autoencoders encode each data sample directly into latent space, the encoding of VAEs involves randomness based on a Gaussian distribution to generate new data instances similar to the training data set.

The model training of the YouTube Mirror project has two steps: intra-modal association and cross-modal association. In the intra-modal training phase, an image VAE and a sound VAE are trained with a set of image data and a set of corresponding sound data extracted from the same videos, respectively. An associator is trained in the cross-modal training phase with pairs of the images and corresponding sounds by using two VAEs trained in the previous phase. The associator is also a VAE, but the input and output data for the associator are the latent representation of the original data. The goal of the associator is to encode the latent space of the image VAE into the latent space of the sound VAE, or vice versa. For example, the associator that is trained with the encoder of the image VAE and the decoder of the sound VAE can generate a sound from a

given input image. The image encoder produces a latent vector from the input image. Then, the associator maps the latent vector for the image to the latent vector for the sound VAE. The sound is reconstructed from the latent vector for the sound VAE via the sound decoder. The reconstructed sound here is a mel-spectrogram. Whereas a spectrogram is the time-frequency representation of a sound on a linear scale, a mel-spectrogram is based on the mel-scale that is analogous to human hearing [16]. At the final stage, the mel-spectrogram is transformed into the time-amplitude representation of the sound that can be heard. The latent vector for the image is also used to produce a reconstructed image via the image decoder of the image VAE.

One of the significant characteristics that VAEs have compared to GANs is that the generated results of VAEs tend to be blurry [17]. As humans' memories, in general, become blurrier over time, this project utilizes the blurry nature of VAEs to create an abstract representation simulating my unconscious understanding of audio-visual associations.

Interactive Audiovisual Mirror

YouTube Mirror aims to let the audience see themselves through the machine-generated images and sounds. For this goal, as its name alludes, YouTube Mirror represents the machine's output in the form of an interactive mirror that can be found in digital media arts, such as Daniel Rozin's *Wooden Mirror* (1999) [18], Kyle McDonald's *Sharing Faces* (2013) [19] and Gene Kogan's *Cubist Mirror* (2016) [20].

Figure 3. An interactive model test with a camera

As shown in Figure 1, the YouTube Mirror installation uses a webcam attached at the top of a portrait-mode LCD monitor to capture images of the audience in real time and represents the reconstructed images of the audience via the monitor. With this configuration, YouTube Mirror is designed to enable the audience to interact with the work as they explore how the reflected images on the monitor change. In addition, one unique aspect of YouTube Mirror as an interactive mirror is that the audience can hear generated sounds that respond to their images. The audiovisual representation of YouTube Mirror as an interactive mirror provides the audience with a multisensory experience that requires active engagement to navigate audiovisual output given in real time.

Figure 3 is a screenshot taken during interactive model testing, showing a generated visual result responding to the captured image of a camera at the top right corner.

The video below is the video documentation of YouTube Mirror.

Visit the web version of this article to view interactive content.

Video 1. The video documentation of YouTube Mirror

Implementation

Data Processing and Model Training

All processes from data collection and model training were conducted with Google Colab[5], an online Python environment that runs on the web browser. The video data for YouTube Mirror was stored in a Google Drive cloud storage that can be used in Google Colab.

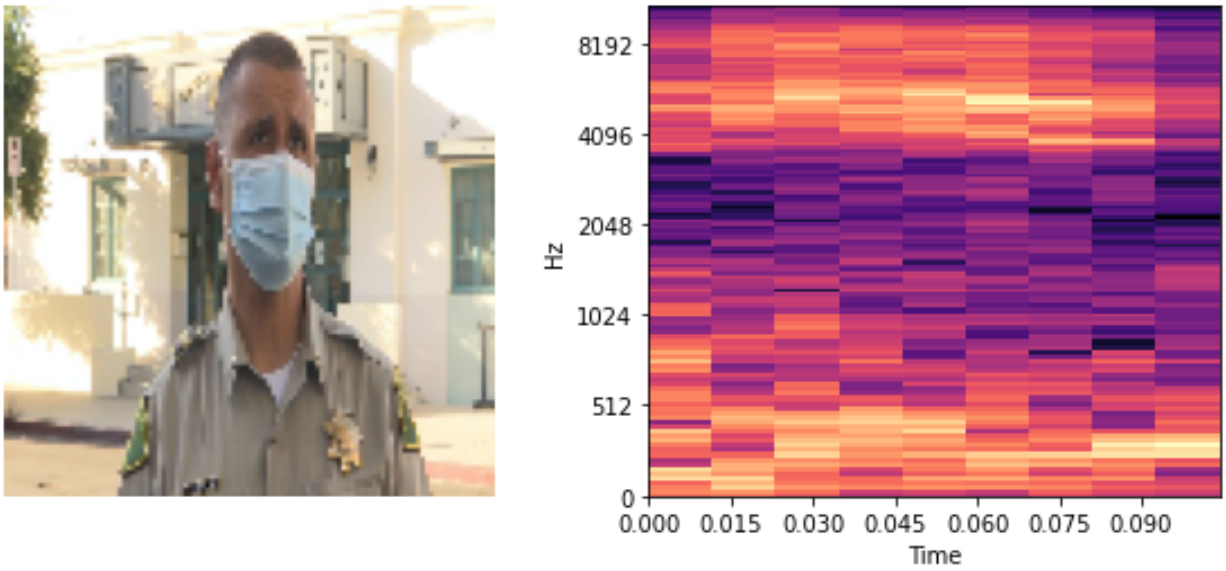

Figure 4. An example of a representative image frame of a video segment (left) and a mel-spectrogram of the same segment (right)

For all 6,091 videos, a 100-millisecond sound segment of each video was extracted. There are two reasons why 100-millisecond segments: First, a tradeoff between latency and accuracy. Although a model that was trained with the full 1-second sounds gave a better result, there was a significant delay in the sound reconstruction process, which is not desirable in making an interactive installation that needs to respond to real-time input. Second, 100-millisecond sounds can be dealt with on a microsound time scale in which humans can perceive an acoustic event [21].

The sound segment was converted into a mel-spectrogram by using the Python audio processing package, *librosa*[6] and its function `librosa.feature.melspectrogram` with the parameters of 512 hop lengths and 1,024 FFT lengths at a sampling rate of 44,100 Hz. As a result, each converted mel-spectrogram has 9 frames of which each has 128 Mel bands. A representative image frame of each video was selected based on the strongest onset event detection [22]. The size of the image was then reduced to 112 pixels in width and 112 pixels in height. Figure 4 shows an example of a pair of a representative image frame and a mel-spectrogram, extracted from the same video segment.

The model training of YouTube Mirror was conducted by using TensorFlow[7], an open-source Python library for machine learning and artificial intelligence. The model architecture of YouTube Mirror was implemented based on an architecture proposed by Jo et al. [15], which used Pytorch[8]. The dimensions of latent vectors in both image and sound VAEs were set to 128. RMSProp (Root Mean Square Propagation) [23] was used as the optimizer of the VAEs. The VAEs were trained with a weighted Kullback-Leibler Divergence (KLD) loss (0.0001) and a learning rate of 0.0001. The training for each VAE ran for 1,000 iterations with an early stopping patience of 8. With the trained VAEs, then, an image-to-sound associator was trained for 1,000

iterations along with a KLD weight value of 0.001, a learning rate of 0.001, an early stopping patience of 8, and RMSProp.

Real-Time Audiovisual Generation

YouTube Mirror has two main modules: an inference module created in Python with TINC (Toolkit for Interactive Computation)[9] and a playback module written in C++ with AlloLib[10], a library for interactive multimedia application development, and TINC. Given a real-time camera input, the inference module continuously produces images and sounds reconstructed from a trained cross-modal generative model. The inference module sends the images to the playback module via the communication feature of TINC. And the sounds are sent to the playback module by using *mmap*, a method of memory-mapped file I/O to reduce latency in sound data communication. The playback module displays the received images as a full-screen texture. Since the trained model is designed to generate images with 112 pixels in width and 112 pixels in height, the texture is linearly interpolated to be a full-screen size. The playback module plays sound sample data received by the *mmap* communication method at a sampling rate of 48,000 Hz.

The sound reconstruction process played the most important role in the real-time interactive audiovisual generation. The sound reconstruction process uses the *librosa* function `librosa.feature.inverse.mel_to_audio`, which is based on the fast Griffin-Lim algorithm [24] for approximate magnitude spectrogram inversion. Considering that the algorithm can cause a severe delay according to its parameters being set, the function was set to run for 5 iterations with 5 Mel frames, a hop size of 512, an FFT window size of 1024, and a sampling rate of 48,000 Hz, generating 2048 audio samples (0.042 seconds) per each function call. The generated audio samples being sent to the playback module were stored in a ring buffer. With an audio callback buffer size of 512 at the same sampling rate, the playback module consumed the generated audio samples with a Hann window being applied, at an interval of 0.0432 seconds.

On top of that, I applied post-training quantization to the trained model using the TensorFlow Lite Converter[11]. Post-training quantization is one of the techniques used to reduce model size, latency and cost for inference, with little degradation in model accuracy. The model was quantized to float16.

As a result, the playback module was able to update the images and the sounds at approximately 20 frames per second on a MacBook Pro (Retina, 15-inch, Early 2013) with NVIDIA GeForce GT 650M.

Exhibition

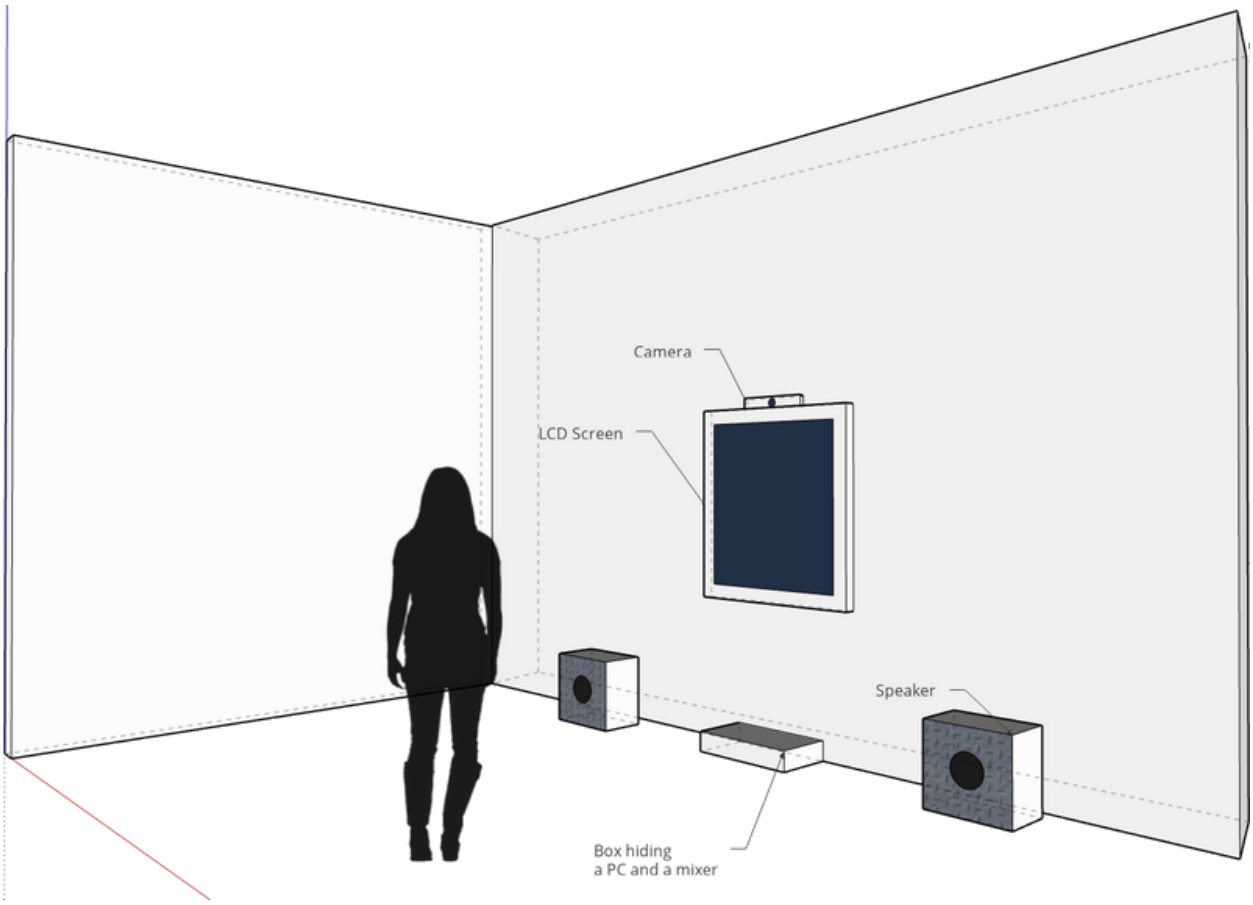

Figure 5. An initial floor plan of YouTube Mirror

As Figure 5 illustrates, a YouTube Mirror installation consists of a camera, a wall-mounted or portrait-mode LCD screen, loud speakers, an audio mixer, and a PC. YouTube Mirror was exhibited as an installation at the Media Arts and Technology Program (MAT) 2022 End of Year Show (EoYS) at the University of California, Santa Barbara (UCSB). The installation used a vertical LCD monitor on a pedestal, as shown in Figure 6.

Figure 6. An installation of YouTube Mirror at MAT 2020 EoYS at UCSB

Due to the use of the camera as input, lighting was one of the most important factors in properly presenting the installation. With too bright lighting conditions, the generated images were too blurry, making it hard to recognize the representation of human bodies. On the other hand, dark lighting conditions caused the representation of the bodies to be filled with only black color.

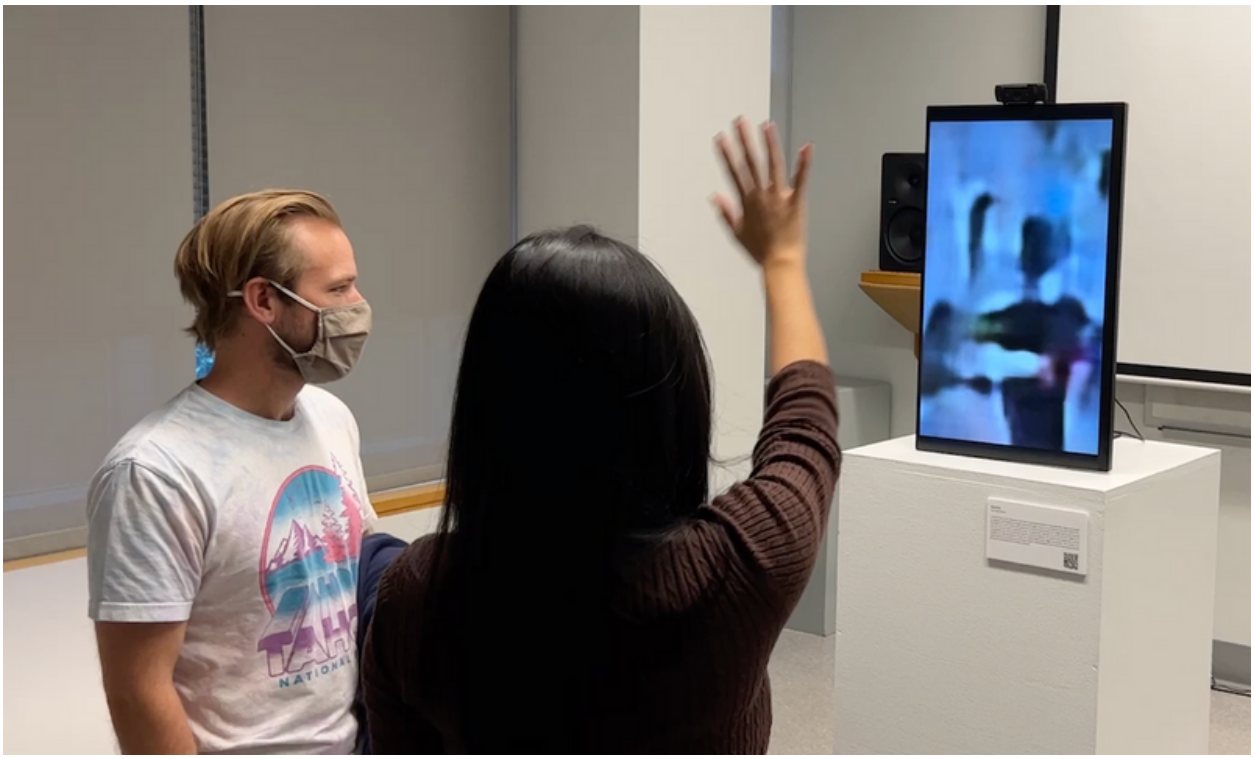

Figure 7. Audience interaction

There are some findings obtained from observing the audience and analyzing a video taken during the exhibition. When they first saw the work, most audiences showed their curiosity about the relationship between the images and the sounds, as well as how the sounds are generated. After realizing that the installation generates the sounds and images in response to the attached camera, the audience exhibited interesting behavior in front of the work. A few audience members stood still, slightly swaying their heads from side to side. As shown in Figure 7, some audiences waved their hands or arms. There was an audience member who waved his arm as if dancing, and a couple of audiences showed acting-like movements.

While some audiences appreciated the abstract visuals generated by the work, others recognized machine-liked sounds that the granularity of the generated sounds brought based on a microsound scale. Some audiences mentioned that the work seems to sonify the images.

Discussion and Future Work

YouTube Mirror is an interactive, audiovisual AI installation based on a cross-modal generative model that learned the implicit relationship between images and corresponding sounds extracted from the same videos that I watched on YouTube. YouTube Mirror is an artistic attempt to simulate my unconscious, implicit understanding of audio-visual association that can exist in the watched videos. YouTube Mirror utilizes the generativity of the machine in creating a unique form of interactive data art.

YouTube Mirror exhibited the participatory nature of an interactive installation. The real-time interactivity of YouTube Mirror led the audience to explore the generative representation of the machine by navigating the latent space of the model with their own body. As a result, the audience showed various movements to interact with the YouTube Mirror installation that provides a multisensory, cross-modal experience.

The YouTube Mirror installation demonstrated potential for utilizing cross-modal generative modeling in interactive audiovisual art. However, there are some limitations that can be further developed and explored. First, the model was trained without any specific categorization of the videos or labeling of the objects that appeared in the videos. This resulted in the generated images being too generalized across the entire video dataset. In general, the human bodies were represented as black unless they had white parts on their clothes. Instead of a fully unsupervised way of training, a better approach to training the model would be to use video data with category labels. Alternatively, exploring different generative modeling architectures, such as diffusion models[25], could prove beneficial.

Second, the generated sounds had no clear distinction according to the given images. Instead, the work produced only machine-like abstract sounds. This limitation can be attributed, in part, to the small lengths of audio samples used during both model training and sound generation. This decision was made to strike a balance between sound quality and inference speed. The concrete sound generation also could be improved with the use of the labeled video data. In addition, exploring alternative methods for generating longer sound samples without introducing severe delays in the sound reconstruction process would be valuable avenues for further investigation.

Third, the YouTube Mirror installation relied on the presence of the audience to differentiate audiovisual output. This somewhat confused the audience to think the generated sound is the result of the sonified camera input. A different installation design could be conceived to alleviate this issue. For example, if there is no audience in front of the installation, the installation can randomly choose representative images from the training data and represent their reconstructed images and sounds. This approach could help the audience understand what the work is attempting to represent even when they do not stand before the camera.

Finally, as an initial attempt to explore possible installation configurations, the current version of the YouTube Mirror installation only uses an image-to-sound associator. The installation can be configured in a different way with a sound-to-image associator. For example, the audience’s sound input can be used to generate corresponding images, allowing them to navigate the model’s latent space with their voice. There is also a possible configuration where both associators are used together so that the audience can see their reconstructed images being modified by or mixed with the images generated from their sound input.

While the original goal was to make a model with my own data, future work could consider personalizing the model with the audience’s data. This could be achieved by simplifying the data processing with a proper user interface to download their YouTube watch history and video data, and streamlining the pipeline for

personalized model training. Moreover, the personalized version of this project would make open-source code available.

Ethical Statement

The development of YouTube Mirror was funded by the Media Arts and Technology graduate program and the Interdisciplinary Humanities Center at the University of California, Santa Barbara. There are no observed conflicts of interest. This paper is partly based on my Ph.D. dissertation [26].

Footnotes

1. https://midjourney.com/ ↩
2. https://takeout.google.com/settings/takeout ↩
3. https://developers.google.com/youtube/v3 ↩
4. https://ytdl-org.github.io/youtube-dl ↩
5. https://colab.research.google.com/ ↩
6. https://librosa.org ↩
7. https://www.tensorflow.org ↩
8. https://pytorch.org ↩
9. https://github.com/AlloSphere-Research-Group/tinc ↩
10. https://github.com/AlloSphere-Research-Group/allolib ↩
11. https://www.tensorflow.org/lite ↩

References

1. Covington, P., Adams, J., & Sargin, E. (2016). Deep Neural Networks for YouTube Recommendations. *Proceedings of the 10th ACM Conference on Recommender Systems*, 191–198. https://doi.org/10.1145/2959100.2959190. Retrieved from https://doi.org/10.1145/2959100.2959190 ↩

2. Zhao, Z., Hong, L., Wei, L., Chen, J., Nath, A., Andrews, S., … Chi, E. (2019). Recommending what video to watch next: A multitask ranking system. *Proceedings of the 13th ACM Conference on Recommender Systems*, 43–51. Association for Computing Machinery. ↩

3. Kingma, D. P., & Welling, M. (2013). Auto-Encoding Variational Bayes. *arXiv:1312.6114 [Stat.ML]*. Retrieved from https://arxiv.org/abs/1312.6114v10 ↩

4. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., … Bengio, Y. (2014). Generative Adversarial Nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems* (Vol. 27). ↩

5. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., & Chen, M. (2022). Hierarchical text-conditional image generation with CLIP latents. *arXiv:2204.06125 [Cs.CV]*. ↩

6. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2021). High-resolution image synthesis with latent diffusion models. *arXiv:2112.10752 [Cs.CV]*. ↩

7. Muller-Eberstein, M., & Noord, N. (2019). Translating Visual Art Into Music. *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 3117–3120. IEEE. ↩

8. Jeong, D., Doh, S., & Kwon, T. (2021). TräumerAI: Dreaming Music with StyleGAN. *arXiv:2102.04680 [Cs.SD]*. ↩

9. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., & Aila, T. (2020). Analyzing and Improving the Image Quality of StyleGAN. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8107–8116. ↩

10. Gatys, L. A., Ecker, A. S., & Bethge, M. (2015). A Neural Algorithm of Artistic Style. *arXiv:1508.06576 [Cs.CV]*. ↩

11. Odlen, I., Verma, P., Basica, C., & Kivelson, P. D. (2020). Painting from Music using Neural Visual Style Transfer. *NeurIPS 2020 Workshop on Machine Learning for Creativity and Design*. ↩

12. Verma, P., Basica, C., & Kivelson, P. D. (2020). Translating Paintings Into Music Using Neural Networks. *arXiv:2008.09960 [Cs.SD]*. ↩

13. Akten, M. (2018). *Ultrachunk*. Retrieved from https://www.memo.tv/works/ultrachunk/ ↩

14. Lee, S. J., & Reeves, T. C. (2007). A Significant Contributor to the Field of Educational Technology. *Educational Technology*, *47*(6), 56–59. ↩

15. Jo, D. U., Lee, B., Choi, J., Yoo, H., & Choi, J. Y. (2020). Associative Variational Auto-Encoder with Distributed Latent Spaces and Associators. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*, 11197–11204. https://ojs.aaai.org/index.php/AAAI/article/view/6778. Retrieved from https://ojs.aaai.org/index.php/AAAI/article/view/6778 ↩

16. Stevens, S. S., Volkmann, J., & Newman, E. B. (1937). A Scale for the Measurement of the Psychological Magnitude Pitch. *The Journal of the Acoustical Society of America*, *8*(3), 185–190. Retrieved from https://doi.org/10.1121/1.1915893 ↩

17. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. Cambridge, Massachusetts: The MIT Press. ↩

18. Rozin, D. (1999). *Wooden Mirror*. Retrieved from https://www.smoothware.com/danny/woodenmirror.html ↩

19. McDonald, K. (2013). *Sharing faces*. Retrieved from https://vimeo.com/96549043 ↩

20. Kogan, G. (2016). *Cubist mirror*. Retrieved from https://vimeo.com/167910860 ↩

21. Roads, C. (2004). *Microsound*. The MIT Press. ↩

22. Bello, J. P., Daudet, L., Abdallah, S., Duxbury, C., Davies, M., & Sandler, M. B. (2005). A tutorial on onset detection in music signals. *IEEE Transactions on Speech and Audio Processing*, *13*(5), 1035–1047. ↩

23. Hinton, G., Srivastava, N., & Swersky, K. (2012). *Neural networks for machine learning lecture 6a: Overview of mini-batch gradient descent*. ↩

24. Perraudin, N., Balazs, P., & Søndergaard, P. L. (2013). A fast griffin-lim algorithm. *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 1–4. ↩

25. Ho, J., Jain, A., & Abbeel, P. (2020). Denoising Diffusion Probabilistic Models. *arXiv Preprint Arxiv:2006.11239*. ↩

26. Park, S. (2022). Data-Driven Audiovisual Art Focused on the Uncertainty in the Human-Data-Machine Loop. *UC Santa Barbara*. ProQuest ID: Park_ucsb_0035D_15599. Merritt ID: ark:/13030/m52g4zj3. Retrieved from https://escholarship.org/uc/item/4773n00c ↩