Machine Learning for the Arts
UCSD SPRING 2019
FINAL PROJECT

This Guitar Does Not Exist

TyMo Guitar Designs:
Tyler Farnan
Mo Rahman

Concept:
The inspiration for this project stems from the fact we are both avid guitarists, and we wanted to apply the skills we gained throughout the quarter with our passions. We were inspired by the websites that use Nvidia's StyleGAN to produce realistic images of objects (such as https://thispersondoesnotexist.com/). With that inspiration, we wanted to generate guitars designs that do not currently exist. Ultimately, our goal is to create realistic novel guitar designs that can be physically built.

Techniques:
The primary technique for this project was implementing transfer learning on a pre-trained StyleGAN to generate guitar images.

StyleGAN Transfer Learning:

Background: In order to use the model that inspired this project, without going through the extensive time and training to build a dedicated model for generating guitars, we opted to use transfer learning on an existing model of Nvidia's StyleGAN that was formerly trained on paintings. Gwern Branwen has created an extensive tutorial on how to accomplish this by using transfer learning with StyleGAN to create anime faces [1]. This technique had the highest likelihood of achieving our goal to create realistic designs for guitars. Another resource was a GitHub repository using the same model and technique to generate portraits [2].

Data Preparation: One prerequisite to using the StyleGAN is that all input images must have a resolution of 512x512 or 1024x1024. Our guitar dataset is made up of images of resolution of 220x220, so we upscaled our images through the use of OpenCV functions to fulfill the requirement, which can be found in upscale.py and remove220.py. Our implementation runs on Google Colab, so we had to download the pretrained model and dataset onto our Google Drive accounts and import them to our notebook from there. StyleGAN does not allow direct passing of arguments to specify the directory a dataset is stored in. Instead, the GAN uses a special filetype known as .tfrecord to specify information about the datasets. That said, directories of images must first be converted into .tfrecord files for StyleGAN to work properly, which can be found in StyleGANGuitarGenerator.ipynb.

Implementation: After all data preparation is complete, simply running the training loop for the StyleGAN to begin training on the dataset is left. This training loop specifies many parameters with specific details about learning rates, generator and discriminator qualities, many of which, were left as default. For the sake of time, we only let the training occur for one epoch, after which, images are instantly generated. The resulting image after one epoch can be seen in the cover image of this report.

Process: Additionally, we trained a deep convolutional GAN from scratch to generate low resolution images and attempted to improve these results with a pre-trained super resolution model. We also ran experiments with style transfer models. These processes and their results are discussed in the next section.

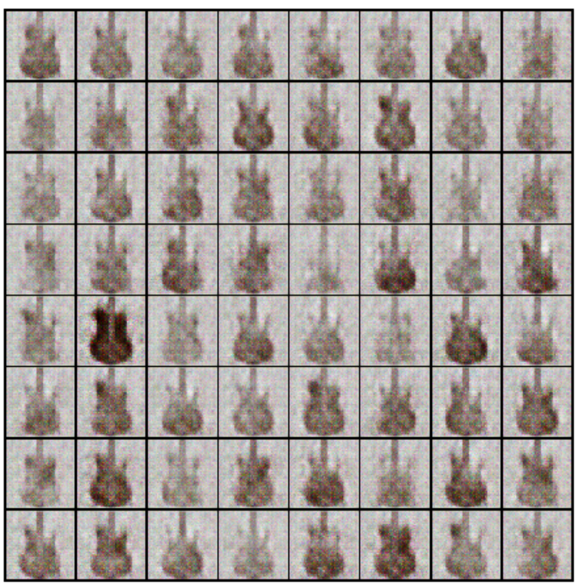
Phase 1: Data Collection

In order to generate new guitar images, we first scraped the (musician's friend and guitar center) to acquire a dataset of stock guitar photos. We acquired around 3,000 guitar images of size 220x220. **Sample Images:**

Phase 2: Basic DCGAN

Our original technique consisted of using a deep convolutional GAN (DCGAN) to generate guitar images. Our first results were very poor greyscale, so we decided to duplicate the dataset to 20k images and retrain.

Results:



Original results with 3k guitar dataset on left, improved results with 20k guitar dataset on right

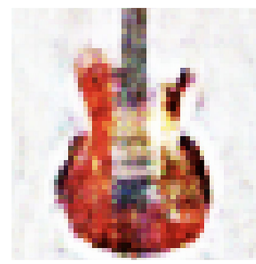
Phase 3: Super Resolution

A simple next experiment was to run the DCGAN generated images through a single image super resolution GAN [3]. After attempting to use a variety of model types and dimension sizes, no positive results were observed. The quality actually reduced in most cases.

Results:

Before

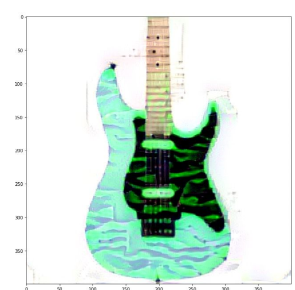
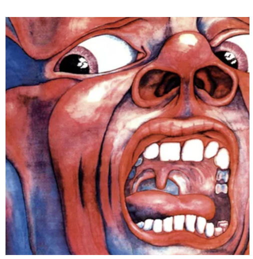
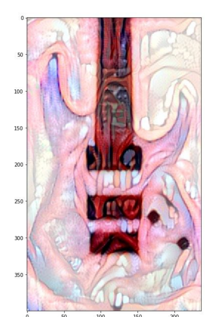
After



Phase 4: Experiments with Style Transfer

Realizing that the DCGAN results may be too poor to restore, we began exploring some basic style transfer capabilities. Below are two basic demonstrations for transferring style of paint design and album art onto guitar bodies. These experiments yielded interesting patterns, but they do not have a very realistic quality.

Results:

Style	**Content**	**Output**
		
		

Phase 5: StyleGAN Transfer learning

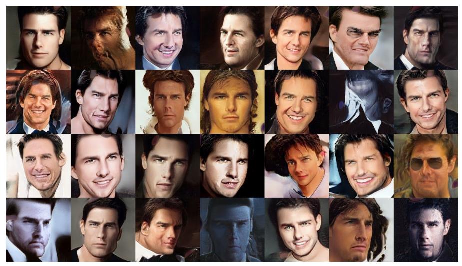
After a bit of exploring, we determined that implementing (transfer learning) on a powerful GAN image generator, could have the most promising results. Specifically, we adapted [2] with our original web-scraped images. This implementation downloads a pre-trained NVIDIA SyleGAN and uses transfer learning to generate novel portraits (see github for sample results). Our first attempts at implementing this repository proved challenging, but ultimately we were able to train a subset of 1,361 guitar images on the portrait model that was originally produced by [2]. After just one epoch of training (3 hours), impressive results were realized:

Final Results:

Considering that these preliminary results were generated from such a small test batch and only a single (3hr) epoch of training, the results are quite impressive. It is interesting to note that there are only a few dominant body styles, with some common features appearing across them. This is likely due to the small training set. In addition, there are some severely warped images, which may be due to a lack of training epochs. Finally, we can observe a peculiar portrait like “style” to these images. This is because the StyleGAN we used had previously been updated with transfer learning on the portraits dataset.

<u>Reflection</u>:

Overall, the quest to generate novel guitar images was a fun and educational experience. We gained experience with a wide variety of generative models, learning their limitations, behavior, and implementation workflows. One of our greatest lessons learned was to be thorough and rigorous when adapting an existing architecture to your own problem. For example, ensuring that data dimensionality and file links are updated. This lesson will be forever saved in our minds, thanks to the Tom Cruise surprise we received after hours of training on an old directory path.



Future Work:

Looking to the future, TyMo guitars sees serious potential to further enhance GAN's for novel guitar generation. Our next steps would be to implement transfer learning with both a larger dataset and longer epoch times. It will be interesting as well to test the effects of different pre-trained styleGANs. An example of a next research question could be: Is theFacesGAN or the CarGAN better suited for transfer learning guitar images? Based on the results of other styleGANs, it seems well within the range of possibilities to create a website where people can request to build GAN generated guitars!

REFERENCE:
[1] Gwern Branwen, *Making Anime Faces with StyleGAN*, February 4, 2019
https://www.gwern.net/Faces#transfer-learning
[2] ak9250, *StyleGAN-Art*, April 18, 2019
https://github.com/ak9250/stylegan-art
[3] zsdonghao,Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network, June 15th, 2017
https://github.com/tensorlayer/srgan

CODE: (https://github.com/ucsd-ml-arts/ml-art-final-tymo)

RESULT: The result images can be found throughout the report as well as on the GItHub repository linked above.