ART AND MACHINE LEARNING
CMU 2022 SPRING
PROJECT 2

Animal Crossing Villager Generator

Group 19
Melina Imamura (School of Computer Science, Senior in CS, concentration in Security & Privacy, minor in Linguistics)
Asumi Gibeau (Mellon College of Science, Senior in Biological Sciences)
Anthony Ramirez (Mellon College of Science, Senior in Mathematical Sciences, minor in Creative Writing)

Description

*Animal Crossing: New Horizon*s is a game where you can have many unique villagers visit your island. They have diverse designs, but generally can be described as anthropomorphic animals. This project is an application of the DCGAN framework to generate new *Animal Crossing* villagers given a data set containing over 2000 pictures of each of the villagers from *Animal Crossings: New Horizons*. After some trial and error, we were able to produce low resolution images that captured what looked like animal crossing villagers from a distance, and contained animal crossing esque eyes and body shape.

Concept

Animal Crossing is a game that is very dear to many of the people in our group. The character designs are very cute and although it is easy to look at a villager and recognize it as a character from *Animal Crossing*, it is harder to pinpoint what makes *Animal Crossing* villager's designs so recognizable. By generating new unique villagers using pictures of all of the existing villagers as a dataset, it may be possible to look at the generated designs and find their common points in order to determine what makes *Animal Crossing* a game with such charming characters.

Technique

For our project, we decided to use the DCGAN algorithm since we wanted to generate completely new villagers just from pictures of pre-existing characters. The notebook used to generate the villagers is a copy of the DCGAN notebook and the dataset was downloaded off of the *Animal Crossing* wiki Nookipedia using the extension Imageye to extract pictures from the website. We ran a number of epochs training the generator by using the discriminator in order to generate accurate pictures similar to the dataset of *Animal Crossing* villagers.

We have used different datasets of the villagers as well. The first dataset we attempted to use included all images from Nookipdia which had images of all Animal Crossing villagers with different angles, such as front facing and a slight turn. We used a range of epochs from 500 to 3000 to generate the images. The second dataset had a smaller set of around 400 images, which contained all villagers, but only front facing. This dataset was chosen as they did not crop themselves in a manner where part of the head does not show, and ensured that each villager was only shown once. This dataset also ran with epochs ranging from 500 to 3000. One final dataset was used in which the villagers were slightly lower in resolution, and were all square spaced to ensure that the entire body of the villager would be shown and trained on the dataset. The villagers differed in the angles they were facing. The epochs ranged from 1500 to 3000 epochs.

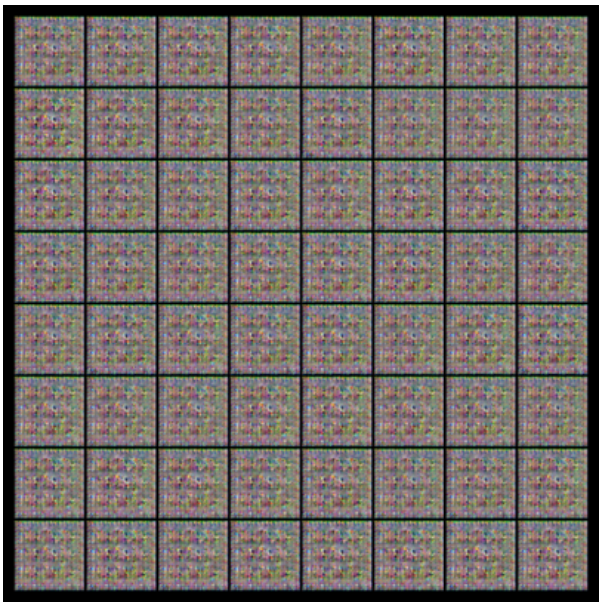
Along with changing the dataset and number of epochs, the number of saved images per number of iterations were also changed. The code originally had it so the generator saved one image per 500 iterations. However, there were cases where we felt that we could have created

better generated images, but the iteration that we wanted would not be saved. Therefore, we changed it so one image was saved every 250 iterations.

Process

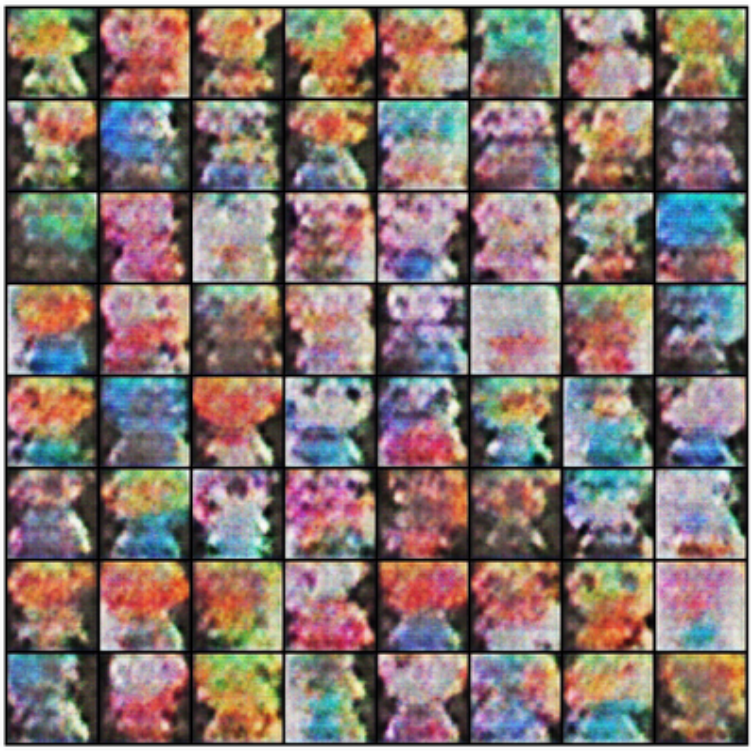
We first gathered images of *Animal Crossing* villagers totalling 2,222 images, put them all in a folder, and tried a run through the whole folder with 5 epochs to test the generator. This resulted in a static image for all fake images, which is reasonable as the generator did not have enough iterations to train from the real images. However, this showed that the code worked, and the generator was able to provide an image.

Sample Images of Dataset



Test Run of 5 Epochs

Since our run with 5 epochs shows that an image is generated, we then attempted 50 epochs to see if it is possible to see a nonstatic image, and determine if the generator was able to produce an image that looked somewhat like a villager. We chose this low number as this was before we were able to use Google Colab Pro. The final image showed that it was producing an image that contained a head, body and eyes, but was extremely blurry. After we were able to access Google Colab Pro, we then decided to run the generator with a large number of epochs.

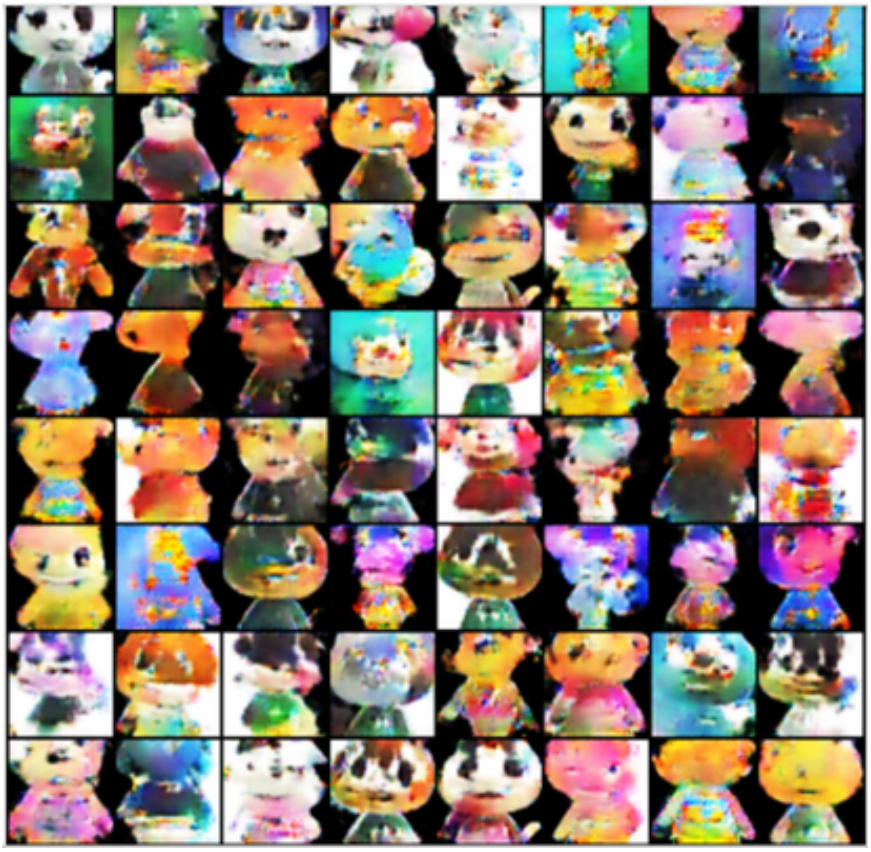


Final Result of 50 Epochs

We tried a run with 5000 epochs using the same dataset of roughly 2000 images, which resulted in a few uncanny villagers. The generator was over trained, and resulted in many of the same images, but was quite high in resolution compared to our other images that we generated. The final result was unsatisfactory, but there were some results near the middle of the training that were a lot more varied and although they were less similar to actual *Animal Crossing* villagers, we felt that they better represented what we wanted the generator to accomplish. Collectively, we decided that 5000 epochs was too many and steered away from such high numbers.

Final Result After 5000 Epochs

It was clear that 5000 epochs was too many, and it appeared that our best results were from the earlier iterations. Therefore, we tried generating images with 800 epochs. These images did not show overtraining, and thus did not show the same few villagers over and over. All images appeared to show defined lines of the bodies of *Animal Crossing* villagers, but the facial features such as eyes and mouths were not clear to see.



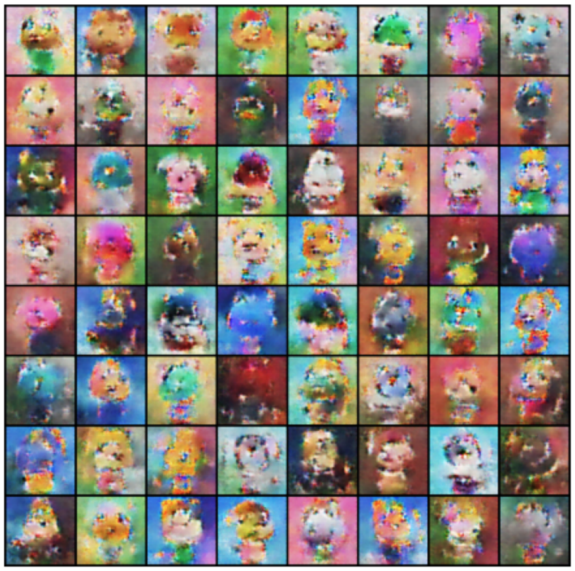
Best Result from 800 Epochs

We tried to narrow down our data set by selecting 400 pictures of villagers that followed a certain pattern: front-facing with a colored background, and running 1500 epochs on them. These results were blurry and a bit unfocused, but they were all unique and mimicked the general style of *Animal Crossing* villagers with big heads, round eyes, and stubby limbs. These images seemed to use more lighter pastel colors and with the colored backgrounds, appeared less harsh than the previous dataset generated images.

Final Result After 1500 Epochs, Real Images on the Left, Generated Images on the Right

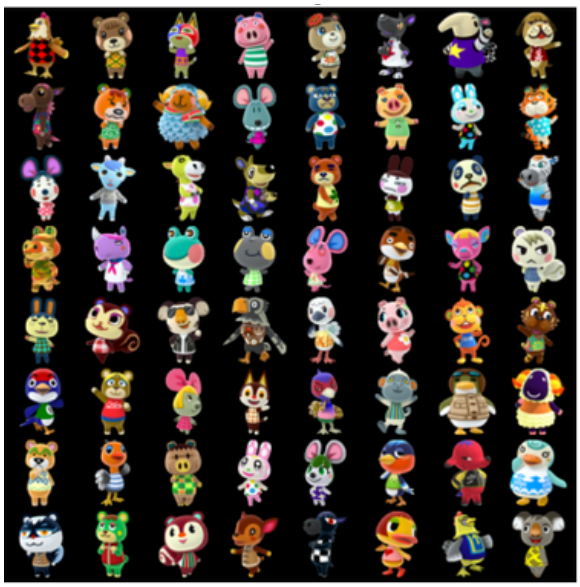
As the final result did not look overtrained, we then attempted to run the same 400 image dataset again but with 3000 epochs. The final result did show that it was overtrained, shown by having a static image. However, we ran this dataset with saving an image every 250 iterations instead of 500, and therefore was able to collect a lot more images throughout the iterations. The inbetween images showed some images where the bodies of the villagers were more distinct in their outlines compared to the final result of 1500 epochs, but the facial features were still blurry. Some of the images also showed what looked like eyes, but others had very blurry faces where even eyes were hard to determine on the image.

Final Result After 3000 Epochs, Real Images on the Left, Generated Images on the Right

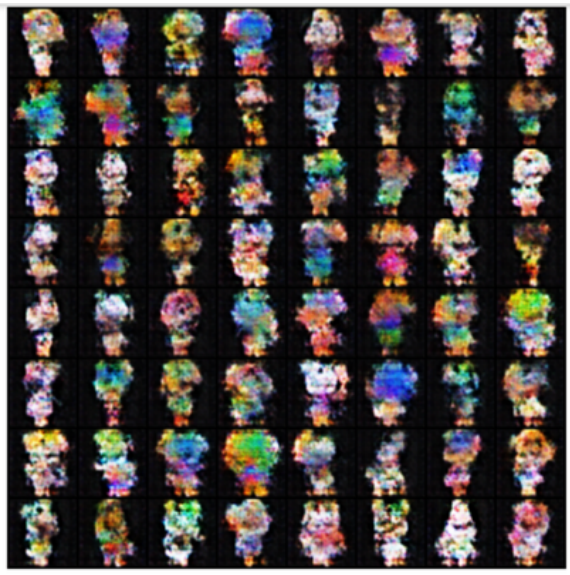


Best Result of 3000 Epochs

Finally, we tried a data set of 400 different full bodied square icons of villagers, and decided to try 1500 epochs again to see if we can create generated images of roughly the same quality as the previous 400 image dataset, but have a full body instead of half of a body. We decided to not continue using this dataset to generate images and thought that previous results were better than any of the generated images that were saved every 250 iterations. As this dataset contained the full bodies of every villager, it seemed that the generator was not able to generate any facial features, as there are no eyes in any of these images, but was also blurry for body features. Although there were legs seen in most images, it is hard to see in most where the head ends and the body begins.



Sample Images of Dataset



Best Result from 1500 Epochs

After all of the images were generated using three different datasets, we compared them to determine which images appeared the most like *Animal Crossing* villagers. Although the images generated from the first dataset had the clearest images, we felt that the colors were too harsh and did not capture the joyfulness of a villager. Therefore, we felt that the best image

Best Image for Project

Reflection

We faced some challenges in generating our villagers due to the limited amount of characters in Animal Crossing. Ideally we would want thousands of pictures to train on, but there are only around 400 villagers in Animal Crossing. After experimenting with different datasets—some with only unique characters and others with multiple pictures of the same characters—we found that having more pictures seemed to create a better result in terms of resolution for our purposes. However, the dataset that had the different colored backgrounds and were all front facing created lower resolution images, but were more vibrant and looked more joyful when compared to the other images generated.

We felt that the final result we chose best reflected what we wanted the generator to accomplish. Even though they were low resolution and had undefined features, the results were all unique and it was clear that they followed the general artistic style of *Animal Crossing*. Of course, it would've been ideal if our final results were high resolution, varied, and reflective of the *Animal Crossing* style, but we felt satisfied with our efforts.

RESULT

This image is from the 400 image dataset that contained front facing villagers, and was the final result after 1500 epochs.

All other images from the training data (final results, results inbetween training, and other test images) can be found in the Google Drive link below.

https://drive.google.com/drive/folders/1zjdGUc_KVPv4AdLjJhjcoe0PHZSX3Z_u?usp=sharing

CODE

The code we ran our datasets on can be seen in the link below:
https://colab.research.google.com/drive/1uKCOK_hLWYL8k2Yu_Htd3pEEKxiAHpbf?usp=sharing