APS360: SYNTHETIC POKEMON

ABSTRACT

This document is for the course APS360 and contains project specifications for a generative neural network that synthesizes new Pokemon. It details the motivations behind the project, some related works, a baseline algorithm, the network architecture, results, and discussion. The following link is the Google Collab link for the project: `https://colab.research.google.com/drive/1aAd4FlFTefnApbFg8SOslUBDhq25EO3I?usp=sharing`

—-Total Pages: 6

1 INTRODUCTION

The goal of the project is to use deep learning methods, specifically Generative Adversarial Networks (GANs) in order to generate synthetic data. That is, to train an image-to-image synthesis model to generate novel samples of a data set of our choice. Our idea is to expand upon one of the suggested projects (Pokemon Classifier), by synthesizing entirely new Pokemon. We will be training the GAN on real images of Pokemon to produce new distinct images from random noise. This presents challenges especially related to data scarcity that will be further discussed and a focal point of the project.

The concept of image to image synthesis (or synthesis in general) has many applications in the field of deep learning which in turn affects fields like healthcare and robotics. For example, generative networks are used to solve the problem of data scarcity in field like healthcare by synthesizing samples to augment the training data. (Sandfort et al., 2019), for example, uses CycleGAN to improve generalizability in CT segmentation tasks by using synthesized non contrast images. In robotics, image synthesis can be done in 3 dimensions to construct 3D models (Xie et al., 2019) which is extremely useful in fields such as mobile robotics where the robot needs to perceive its surroundings and map camera inputs to a prediction and depth-based reconstruction of the objects around it.

In order to illustrate why deep learning is a good approach for these kinds of problems, we use the concept of videos. Videos contain sequences of images in the form of frames that are positioned close to each other temporally depending on the frame rate. Problems such as frame estimation (interpolating extra frames in between) then have more traditional techniques like motion estimation used in video compression (Pesquet-Popescu et al.) in order to interpolate frames. In the context of synthesizing entirely new images, it is much more difficult to extract the required features necessary to solve the problem since there is not an obvious locality/mapping to exploit. Instead, we rely on the available data and attempt to build complex models that can solve this problem. Furthermore, there is the aspect of random noise used to generate the images which also naturally lends itself to machine learning based techniques. In the context of 3D reconstruction, unsupervised deep learning is favourable since it is considered to be an ill-posed problem extensively studied by machine learning and computer graphics communities due to the loss of geometry information when projecting 3D objects onto 2D images (Liu & Han, 2021), thereby making it difficult to perform the reverse.

In summary, the ability to synthesize artificial visual data is a currently relevant field of research in the field of computer vision and lends itself to contributing to possible solutions of complex vision/perception problems in different fields. This project serves as a more grounded way to explore and implement these generative models within the scope of the course in order to gain the exposure and skills needed for the group members to possibly contribute to the more complex ones listed above.

The figure below shows the overall idea for the model through the structure of a generative adversarial network.

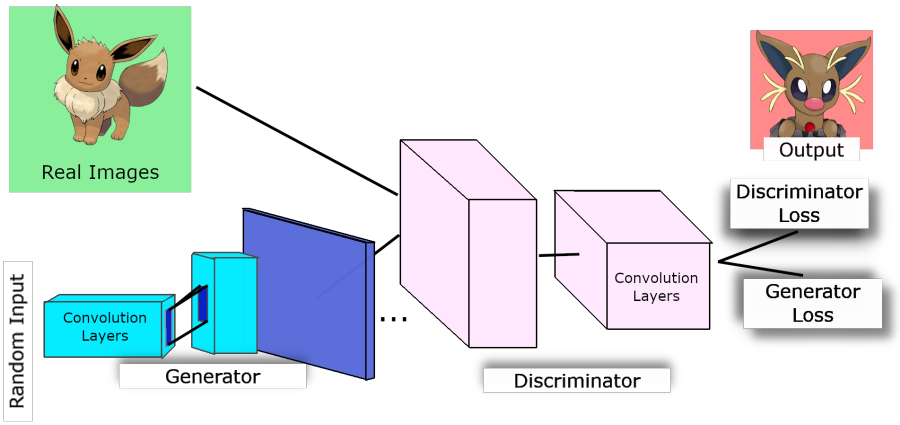

Figure 1: GAN network diagram.

2 Background & Related Work

Along with the aforementioned image synthesis works such as 3D model reconstruction (Xie et al., 2019), and healthcare sample synthesis (Sandfort et al., 2019), there are many similar works involving image-to-image synthesis using GANs. One of the classic examples of this is generating fake faces which many researchers have been working on improving since, yielding state of the art models like StyleGAN (Karras et al.) which is a non-traditional architecture for GANs that borrows the concept of *adaptive instance normalization* from style transfer literature. Some improvements to the architecture include generator normalization, improving regularization, and improving the image quality (Karras et al., 2020).

Considering GAN applications that are more directly related to our project, there is (Yagoub, 2022) which uses Deep Convolutional GANs (DCGANs) in order to generate synthetic ”anime” faces. The model that we implemented follows a similar architecture due to similar problem definitions. In essence, we are also trying to synthesize characters from data and noise which lends itself to using a similar architecture.

A similar project was attempted before as seen in (Lazarou, 2020), where the author also implemented a DCGAN to generate new Pokemon which is the same style of architecture that we used. However, this work differs from ours as this was done 3 years ago, before the new generation of Pokemon games released. The author also used old sprite data which we expanded upon. This work is also useful as a baseline for evaluating our approach.

3 Project Details

3.1 Data Processing

For our data we collected images of Pokemon from various Pokemon related websites. Due to the limited number of Pokemon, and the amount of data needed for the network, a number of preprocessing and augmentation steps were performed on the data. Images were collected from Pokemondb, and Bulbapedia (2023). Both of these sites allowed us to collect two styles of approximately one thousand Pokemon images for use in our network. We also collected ”shiny” - officially recoloured - versions of Pokemon further adding to our data set. To clean the data, images that were of multiple Pokemon were deleted and images missing a white background were given one to be consistent with the rest of the data set. The image values were also normalized to be a value between negative and positive one, to match our network architecture and increase performance.

To preprocess the data, resizing, flipping, and recolouring were done. First, we changed the colour values of each image to make five extra recolours of each Pokemon. This expanded the number of images we had to approximately 8000. To further increase the variation in our data set we also randomly flipped some images. Then, we resized each image to be 3x64x64 to reduce the complexity and training time needed for the network to learn. Other experiments were done to add Gaussian blur and rotation but neither resulted in higher quality output images. The image below shows an example of a Pokemon that has been recoloured for our data set.

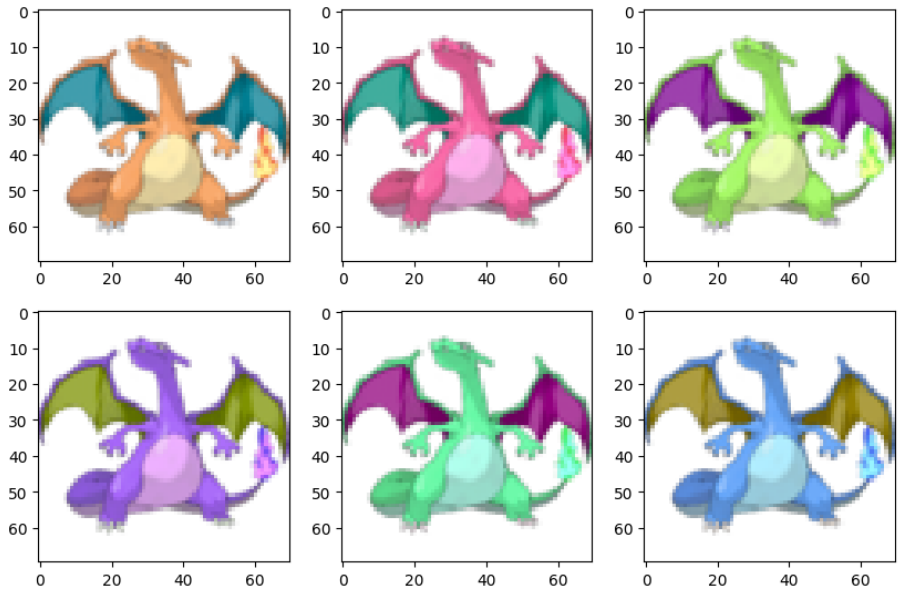

Figure 2: Pokemon Recolouring Preprocessing.

3.2 Baseline Model

To provide a baseline for our model a hand-coded algorithm was created to create images of Pokemon. The model uses the images from our data set to produce new images of Pokemon. The model uses the OpenCV library to add the image values of a handful of Pokemon, then we generate Gaussian noise over top of the image, and then take an average of the image based on how many Pokemon we sampled. Through testing we found a batch of five to ten Pokemon produces a reasonable image. Additionally, another way we can compare our results qualitatively is through the aforementioned older Pokemon AI generator (Lazarou, 2020).

3.3 Architecture

The network that we used is a Generative Adversarial Network, consisting of two convolutional networks acting together: the generator, and the discriminator. Both consist of a sequence of convolutional layers, activation functions, and batch normalizations. However some key differences are needed between the two networks. First, the generator takes noise as input, so the first convolutional layer is based on the size of the noise. As output, the generator produces an image which matches the size of the images in our data set (3x64x64). The discriminator takes in an input of these images as well as images from our data set, and produces a prediction of whether the inputted image is real or fake, a represented by a value between 0 and 1, which was obtained using a Sigmoid function. Lastly, a key change that we found to be beneficial was using larger kernels (4x4) with striding. The figure below shows the sequence of layers and their parameters.

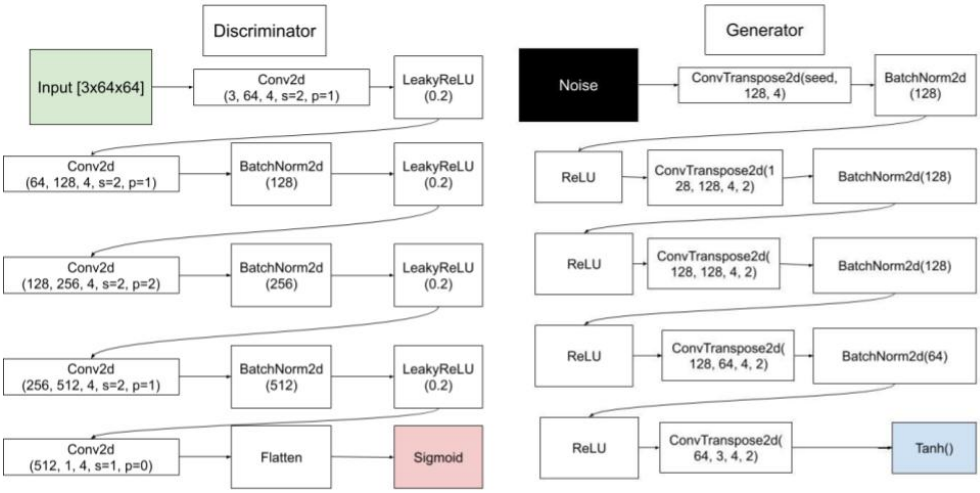

Figure 3: GAN Model Architecture.

4 RESULTS

When looking at the results there are two goals that we want the model to achieve. First, that the images are similar to the real images in the data set, and secondly, that the model does not produce the same output repeatedly, otherwise known as mode collapse. Analysing both the quantitative and qualitative results allows us to evaluate whether or not the model achieves these goals.

4.1 QUANTITATIVE RESULTS

Quantitatively evaluating GANs is not an easy task. It is not possible to simply evaluate the generator using the discriminator, since a good result could either mean that the generator is generating good quality images and fooling the discriminator, or that the discriminator is simply bad at recognizing generated images. Several metrics have been proposed to quantitatively evaluate the performance of GANs, however there is currently no consensus on which is the best for evaluating their strengths and weaknesses. Rather, metrics are chosen depending on the model and the type of data being generated (Borji, 2019). To evaluate the performance of our model, we decided to use the Frechet Inception Distance (FID) which is a score used to estimate the similarity in the features of the generated images when compared to the dataset. The FID score is calculated by measuring the features of the generated data using a pre-trained feature extraction neural network, and comparing this with the features of the real data (Brownlee, 2019a). By finding the FID score of checkpoints in our model's development, we can quantitatively see the improvement of our model. First, comparing a new set of images (which was not trained on) to the training data set yields a score of near zero, which is expected, and means the images are of similar quality. Our initial model resulted in an FID score of 421.5 and through modifications to our model and further testing we decreased our FID score to 210.5. This is still a somewhat high score and indicates that the features in the generated data are still somewhat different from the actual data.

4.2 QUALITATIVE RESULTS

Since a GAN is a generative type of network, the qualitative results can be easily shown by the quality of the images produced by the network. Due to the nature of GANs, the most common and most intuitive way to evaluate a GAN is qualitatively, by visually judging the quality of the images (Borji, 2019). This also allows us to evaluate whether our model is suffering from mode collapse, where the model produces the same output regardless of the inputted noise. If a model undergoes mode collapse then it will produce duplicates which are visually identical. These duplicates will not necessarily be recognized by an analysis software since the pixel values will not be identical, so the best way to test for mode collapse is by visually inspecting the data. A batch of generated images is shown below.

Figure 4: A batch of generated Pokemon.

4.3 Discussion

The quantitative and qualitative results give us a good sense for how our model is performing. By looking at the quantitative results, we can see that the images have improved in quality over the course of improving our model, yet they are not yet at the same quality of an official Pokemon image. As for the qualitative results, there are a number of aspects that can be discussed by looking at the output images. By looking at the output we can determine that the model is performing well, and we can clearly see the shape of the newly generated Pokemon, each output is distinct from the other (there are no duplicates), and that generating a new output produces different results. We can conclude that the model as learned how to create the shapes and colours of Pokemon, however not the inner details, such as arms, legs, eyes, and other bodily shapes. As these details are extremely varied over the different types of Pokemon we did not expect the model to show these details due to the limited amount of available data.

5 Ethical Considerations

A few ethical limitations need to be considered regarding copyright infringement during both generating images and the usage of training data. When generating new images we must consider the copyright of the new images produced. There has been recent controversy surrounding AI generated art and its place in copyright law. It can be determined that AI art does not have copyright protection (Mathur, 2022). The use of copyrighted material as training data may also violate some ethical boundaries, In Canada however, though our neural network has a chance of reproducing copyrighted

material, under the ”Fair Dealing” exception we can avoid this issue due to the reproduction being for research purposes (Canadian-Government).

6 PROJECT DIFFICULTY

While the difficulty of the project is subjective the project was certainly more difficult than expected and required a number of techniques to complete. The project was more difficult because:

- **The network type went beyond the scope of the labs.** The labs in the course covered a number of networks, however they mainly dealt with classification networks. As our project was a GAN, not only did it require two networks, but also it was of a generative type as well.

- **Limited data and data processing.** Due to the limited amount of existing Pokemon, our data was extremely limited, because of that, we augmented and processed our data through recolouring and resizing, as well as experimenting with translations, rotations, and reflections.

- **Long training times and multiple failure modes.** As is the problem with GANs, our model took a long time to train and iterated over many epochs. Additionally, interpreting the losses of the generator and discriminator requires research and understanding of how the networks are interacting. There can be many reasons for the network failing and understanding the cause of mode collapse, convergence failure, or random guessing is difficult (Brownlee, 2019b).

- **Volatile Network.** Because of the complexity and length of training, the network is sensitive to changes in hyperparameters. The GAN relies on a balance between the generator and the discriminator, so optimizing hyper parameters to maintain said balance is time consuming, and often times the incorrect hyperparameters can cause the model to collapse.

- **Optimizing two networks.** Since our GAN uses two networks, one in the generator, one in the discriminator, we needed to experiment with different activation functions, layer ordering, and kernel sizing. As such optimizing both networks took lots of time and research.

- **Lack of standardized testing metric.** Due to the generative nature of GANs, there is no standardized metric that is used to test them. The most common evaluation method involves visual inspection by humans to determine the quality and variety of generated images (Borji, 2019). However, receiving human feedback is very time- and resource-expensive and there is a lot of variance on human opinion, so a large sample size is needed to obtain usable information. This rendered this form of testing outside the scope of our project, so we had to rely on other testing metrics to analyze our results.

REFERENCES

Ali Borji. Pros and cons of gan evaluation measures. 2019.

Jason Brownlee. How to implement the frechet inception distance (fid) for evaluating gans. 2019a.

Jason Brownlee. How to identify and diagnose gan failure modes. 2019b.

Bulbapedia. List of pokémon by national pokédex number. https://bulbapedia.bulbagarden.net/wiki/List_of_Pok%C3%A9mon_by_National_Pok%C3%A9dex_number, 2023.

Canadian-Government. Copyright – learn the basics protect your original works. learn why copyright matters. https://ised-isde.canada.ca/site/canadian-intellectual-property-office/en/copyright-learn-basics/copyright-learn-basics-protect-your-original-works-learn-why-copyright-matters.

Tero Karras, Samuli Laine, and Timo Aila. Stylegan.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. 2020.

Conor Lazarou. I generated thousands of new pokemon using ai. 2020.

Zechen Liu and Zhihua Han. Efficient uncertainty estimation for monocular 3d object detection in autonomous driving. 2021.

Atreya Mathur. Art-istic or art-ificial? ownership and copyright concerns in ai-generated artwork. *Center for art law*, 2022.

Béatrice Pesquet-Popescu, Marco Cagnazzo, and Frédéric Dufaux. *Motion Estimation Techniques*. Telecom ParisTech.

Pokemondb. https://pokemondb.net/pokedex/shiny.

Viet Sandfort, Ke Yan, Perry J. Pickhardt, and Ronald M. Summers. Data augmentation using generative adversarial networks (cyclegan) to improve generalizability in ct segmentation tasks. 2019.

Haozhe Xie, Hongxun Yao, Shengping Zhang, Shangchen Zhou, and Wenxiu Sun. Pix2vox++: Multi-scale context-aware 3d object reconstruction from single and multiple images. 2019.

Nassim Yagoub. Gan anime faces. https://www.kaggle.com/code/nassimyagoub/gan-anime-faces, 2022.