Analyzing Wasserstein Generative Adversarial Networks with Gradient Penalties

CS665 Final Project

JOSHUA DAUGHERTY, The University of Alabama at Birmingham, USA
HAMID CHOUCHA, The University of Alabama at Birmingham, USA
VAISHAK MENON, The University of Alabama at Birmingham, USA
ANTHONY NETTLES, The University of Alabama at Birmingham, USA

Generative Adversarial Networks are at the forefront of merging game theory with machine learning, offering a novel approach to data generation and manipulation in digital environments. By setting up a competition between two neural networks – a ***generator*** and a ***discriminator*** – GANs learn to produce synthetic data that closely resembles real-world examples. This innovative technique has applications across a large variety of domains, including image and text generation, data augmentation, and anomaly detection. The transformative potential of GANs in reshaping data-driven decision-making systems proves the significance of their study and application in modern machine learning.

CCS Concepts: • **Computing methodologies** → **Neural networks**; **Unsupervised learning**; **Model development and analysis**.

Additional Key Words and Phrases: Neural Networks, Unsupervised Learning, Generative Networks, Game Theory, Machine Learning

ACM Reference Format:
Joshua Daugherty, Hamid Choucha, Vaishak Menon, and Anthony Nettles. 2025. **Analyzing Wasserstein Generative Adversarial Networks with Gradient Penalties**: *CS665 Final Project.* 1, 1 (February 2025), 7 pages.

1 INTRODUCTION

Generative Adversarial Networks (GANs) blend machine learning with strategic thinking, reshaping how we create and manipulate data. At the intersection of computer science and artificial intelligence, GANs mimic real-world data distributions by having two networks pitted against each other: one generates data while the other evaluates its authenticity. This approach offers fresh insights into data generation, augmentation, manipulation, making it highly relevant in today's digital landscape.

In our paper, we analyze our own implementation and creation of GANs, aiming to understand their complexities and applications in modern machine learning. By connecting the theory we study to practical examples, we uncover the potential of GANs across various fields, from academia to industry. Through an examination of their development, challenges, and real-world uses, this paper aims to provide a comprehensive understanding of GANs and their impact on data-driven decision-making in today's increasingly digital era.

Authors' addresses: Joshua Daugherty, The University of Alabama at Birmingham, Birmingham, USA, joshuadaugherty@acm.org; Hamid Choucha, The University of Alabama at Birmingham, Birmingham, USA, hamidc@uab.edu; Vaishak Menon, The University of Alabama at Birmingham, Birmingham, USA, vmenon19@uab.edu; Anthony Nettles, The University of Alabama at Birmingham, Birmingham, USA, arnet@uab.edu.

2 DATASETS USED

In our testing, we utilized several datasets that are commonly used in computer vision and machine learning tasks. These datasets helped us evaluate the performance of our models in generating realistic images and solving various image-related tasks. Below, we describe the two primary datasets used in our work: CelebA and Animals-10.

2.1 CelebA

The Celebrity Attributes (CelebA) dataset is used in computer vision and machine learning, with over 200,000 celebrity images, with 40 attribute labels per image. These annotations make it valuable for tasks such as image generation, attribute prediction, and data augmentation. Researchers utilize GANs trained on CelebA to generate realistic face images with diverse attributes, while also employing it to train models for attribute prediction tasks like gender detection and facial expression analysis. However, the dataset also raises ethical concerns regarding privacy and consent, given its use of celebrity images.

2.2 Animals-10

We also used the Animals-10 dataset from Kaggle, which is composed of roughly 28,000 animal images from 10 classes. Using this library, our models can be trained for tasks like classification and object detection for applications such as wildlife monitoring and species identification. Additionally, the dataset facilitates data augmentation in lieu with our GAN, which can enhance model performance and generate synthetic images for diverse applications. Efforts to expand the animal dataset can further improve its usage for advancing research in fields like wildlife conservation, veterinary medicine, and ecology.

3 MODEL CREATION

This section will review the construction of a generalized GAN and explain what distinguishes a Wasserstein Generative Adversarial Networks (WGAN) from a typical GAN. We will also discuss key components that define each type of model, such as the generator, discriminator, and the loss functions used during training.

3.1 General GAN Construction

The creation of a Generative Adversarial Network involves the construction and training of two neural networks: the **generator** and the **discriminator**. The generator takes random noise as input and generates synthetic data samples, while the discriminator distinguishes between real data from the training dataset and fake data generated by the generator. During training, the generator and discriminator engage in a two-player mini-max game, where the generator aims to produce data that is indistinguishable from real data to fool the discriminator, while the discriminator aims to accurately classify between real and fake data. This adversarial process drives both networks to improve iteratively, with the generator learning to generate increasingly realistic data samples, and the discriminator becoming better at distinguishing between real and fake data.

3.1.1 ***Training.*** The training process involves optimizing the parameters of both the generator and discriminator networks using respective loss functions. The generator's loss function encourages it to produce data that the discriminator classifies as real, while the discriminator's loss function penalizes incorrect classifications of both real and fake data. This iterative training process continues until a certain convergence criterion is met, such as reaching a predefined number of training epochs or achieving a satisfactory level of performance. At convergence, the generator

ideally produces synthetic data that closely resembles the real data distribution, and the discriminator is unable to distinguish between real and fake data with high confidence.

3.2 Wasserstein Generative Adversarial Networks

Wasserstein Generative Adversarial Networks (WGANs) take a different route from traditional GANs by using the Wasserstein distance for optimization. This change helps tackle some common problems in typical GANs, like mode collapse and unstable training. By focusing on reducing the Wasserstein distance between real and generated data distributions, WGANs offer more stable training and higher quality sample generation, which is especially useful for tasks like creating images and augmenting data.

Additionally, using the Wasserstein distance ensures that the training process maintains meaningful gradients, addressing issues like gradients disappearing during training. However, setting up WGANs requires careful adjustments of parameters like the Lipschitz constraint and learning rate to get the best results. Despite these challenges, WGANs are a significant step forward in generative modeling and offer a more stable approach to training these type of models that create synthetic data.

3.2.1 ***Wasserstein Distance.*** The Wasserstein distance measures how much "work" it takes to change one probability distribution into another. In GANs, it shows how different the distribution of real data is from the distribution of generated data. Unlike simpler distance measures like Euclidean distance which look at specific points, the Wasserstein distance takes into account the overall shapes of the distributions.

3.2.2 ***Gradient Penalties.*** Gradient penalties are an important method used in WGANs (also called WGAN-GPs) to control how much the discriminator's predictions can change. This helps to prevent issues like mode collapse, where the generator focuses on only a few samples. In WGAN-GPs, the gradient penalties work by adding an extra part to the discriminator's loss function that punishes large changes in its predictions. This encourages the discriminator to keep its predictions consistent throughout different parts of the data, making the training smoother and more generalized. By using gradient penalties, WGANs can improve their training stability and produce better results.

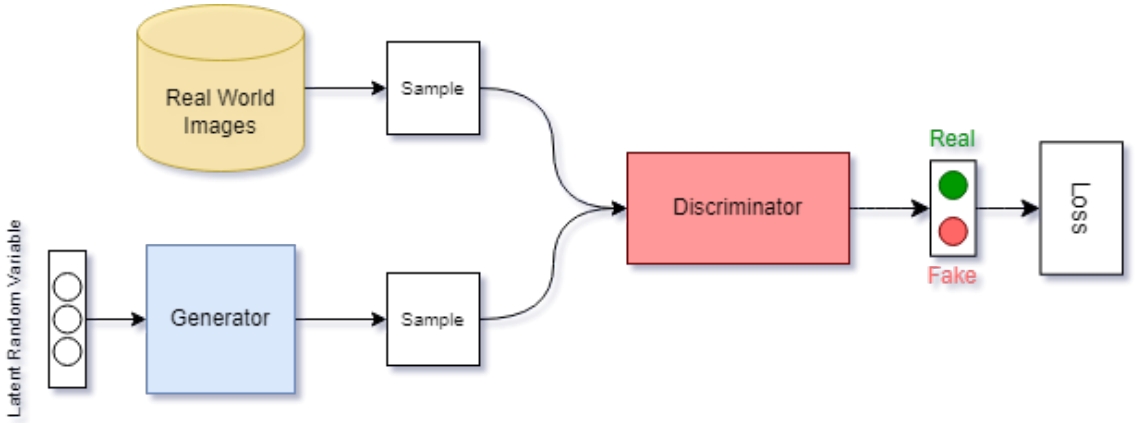

Figure 1. A visual simplification of a typical GAN/WGAN layout, showing data-flow and feedback

4 IMPLEMENTATION

We will now go more in depth with regard to our techniques, functions, and algorithms used.

4.1 Tuning Hyper-parameters

A key aspect of optimization for machine learning models is the fine-tuning of hyper-parameters, and WGAN-GPs are no different. We adjusted things like Lipschitz Constraint, Learning Rate, Batch Size, and more.

4.1.1 ***Lipschitz Constraint.*** Adjusted the coefficient λ to balance the strength of gradient penalty enforcement for controlling discriminator gradients. Higher λ values increase the penalty strength, but also slowed training down.

4.1.2 ***Learning Rate.*** Experimented with different learning rates to find the optimal value for stable training and convergence speed, considering gradual decay over time for improved performance.

4.1.3 ***Batch Sizes.*** Tested different batch sizes to optimize gradient estimation accuracy and training stability while considering trade-offs between convergence speed and memory usage. Smaller batch sizes provided us with more stable training but also slowed down convergence rates.

4.2 Optimizations

For the first model we created and trained on the Animals-10 dataset, we initially used a linear classification technique, with a Leaky ReLU activation function, for both the generator and discriminator. From this, we mostly received outputs of blobs of colours (intended to be the animals), and some not-so-good feedback from the discriminator. We then swapped over to the CelebA dataset and began using a model with convolution layers in 2-dimensions, which produced much better results.

4.2.1 ***Activation Functions.*** As mentioned before, we tested a few different activation functions - ReLU, Leaky ReLU, Sigmoid, and Tanh. We also tested using different functions for each of the generator and discriminator, using ReLU and Leaky ReLU, respectively.

4.2.2 ***Wasserstein Loss Function.*** After switching from Binary Cross Entropy loss (BCEloss) to the Wasserstein loss function, both the generator and discriminator showed improvements, with generated images becoming more realistic. Below is a pseudocode-style implementation of the WGAN loss with gradient penalty (WGAN-GP).

```
    fake_data = generator(z)

    d_real = discriminator(real_data)
    d_fake = discriminator(fake_data)

    d_loss = mean(d_fake) - mean(d_real)
    g_loss = -mean(d_fake)

    gp = compute_gradient_penalty(real_data, fake_data)
    d_loss += gp

    update_discriminator(d_loss)
    update_generator(g_loss)
```

Figure 2. Wasserstein loss calculation (pseudo-code)

4.2.3 ***Gradient Penalties.*** Implemented gradient penalties in WGAN-GP to enforce the Lipschitz constraint on the discriminator, adjusting the coefficient λ to balance penalty strength and training speed. This regularization technique stabilized training, encouraged smoother convergence, and improved the quality of generated samples even further.

5 RESULTS

Here we will introduce our preliminary and most recent outputs (focusing solely on the CelebA model).

5.1 Preliminary Results

For these results, we had a *learning rate* that we came to understand was a bit too low, and neither network was really understanding what a good image looked like.

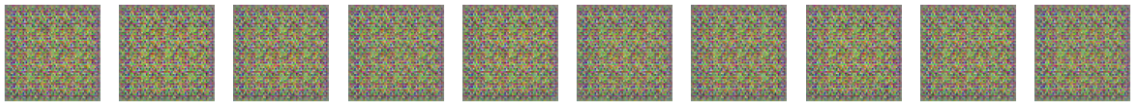

Figure 3.1. As you can see, this is not exactly a person.

Figure 3.2. We increased the learning rate and got some images that look a bit more familiar, yet...not quite right.

5.1.1 ***Losses.*** Below are example loss outputs seen after preliminary testing:

$$[1400/1583]\ Loss_D : 0.7767\ Loss_G : 3.3145\ D(x) : 0.9457\ D(G(z1)) : 0.4450\ D(G(z2)) : 0.0596$$
$$[1550/1583]\ Loss_D : 1.0853\ Loss_G : 2.6870\ D(x) : 0.5702\ D(G(z1)) : 0.2489\ D(G(z2)) : 0.1005$$

5.2 Recent Results

After the addition of the Wasserstein loss function and gradient penalties, we saw vast improvements and some very impressive outputs. Here are some of the best outputs and cherry-picked examples of good images.

Figure 4.1. While these faces are more distinct and unique, they may not be very well-defined or distinguishable from surroundings.

Figure 4.2. These WGAN-GP outputs are much more recognizable. Although low resolution, these are synthetic images of people with various smiles, hair, and facial orientations, and fairly distinguishable image backgrounds.

6 POTENTIAL APPLICATIONS

After reviewing the theory, discussing the model, and seeing sample synthetic images generated by our WGAN-GP, let us take a look into some areas where a WGAN-GP, and GANs in general could be, and likely already are being used.

6.1 Image Generation

Probably the most obvious field of application, sufficiently trained WGAN-GPs can generate high-quality realistic images of things from human faces, to animals, to whatever it may be trained on. These models and their outputs can be used to validate authenticity of artwork, create synthetic data for use in research and industry, or even train WGANs.

6.2 Data Augmentation

Somewhat following from section 6.1, WGAN-GPs can help in augmenting existing datasets by producing additional images to be used in training or testing. This augmented data can enhance the performance and robustness of machine learning models trained on tasks like image classification, object detection, and facial recognition.

6.3 Anomaly Detection

By learning the distribution of normal data during training, deviations between generated samples and the learned distribution would indicate anomalies. For example, a WGAN-GP trained on animals could assist in detecting anomalies in wildlife monitoring images. If during monitoring, an image is significantly different—perhaps containing an unexpected or rare animal species—it could be flagged as an anomaly. This capability aids conservation efforts by automatically identifying unusual sightings or potential threats to biodiversity without manual inspection of each image.

6.4 Art and Creativity

A bit on the more general populous-facing side of things, the generated images can serve as a source of inspiration for artists and designers, providing them with an array (wordplay intended) of high-quality images of human faces, animals, and other subjects for creative projects such as digital art and multimedia productions.

6.5 Research and Education

Back into the field of academia, researchers and educators can leverage trained WGAN-GPs and their generated images for various educational and research purposes. For instance, it can be used to study human perception, animal behavior, or the impact of facial expressions on emotional recognition.

6.6 Privacy Protection

One area of concern in using the Celebrity Attributes dataset is that of privacy, consent, and anonymity. Synthetic images generated by the WGAN-GP can be used to protect individuals' privacy by replacing real images with realistic but synthetic ones. This approach allows researchers and organizations to share and analyze sensitive data without compromising individuals' privacy.

6.6.1 Medical Confidentiality. Synthetic images generated by the WGAN-GP can protect patient privacy in medical datasets. This technique enables researchers to share and analyze sensitive medical imaging data without compromising patient confidentiality. Additionally, synthetic images can simulate diverse medical scenarios for developing diagnostic tools and treatment strategies, while still respecting privacy concerns.

7 CONCLUSION

In essence, Generative Adversarial Networks and their variants like Wasserstein GANs offer powerful solutions for data generation and manipulation in digital environments. By having neural networks compete against one another in adversarial competitions, these models learn to create synthetic data that closely resembles real-world examples. The exploration of GANs' capabilities reveals their potential applications across various domains, including data augmentation, image synthesis, and anomaly detection. However, challenges such as training stability, mode collapse, and ethical considerations persist, requiring ongoing research and innovation to address.

Navigating the landscape of GANs explains their relevance in today's technologically advanced era, where cheap, realistic, and often times synthetic, data is increasingly in demand. GANs expand traditional machine learning approaches by introducing adversarial training mechanisms, enabling the generation of high-integrity data representations. This not only opens up new possibilities for data augmentation and image manipulation but also fosters innovation in fields such as computer vision, natural language processing, and biomedical research. Despite remaining challenges, GANs continue to shape the future of data generation and manipulation.

REFERENCES

[1] H. Petzka, A. Fischer, and D. Lukovnicov, "On the regularization of Wasserstein GANs," [stat.ML], 2018. https://arxiv.org/abs/1709.08894

[2] A. Mallasto, G. Montúfar, and A. Gerolin, "How Well Do WGANs Estimate the Wasserstein Metric?" [cs.LG], 2019. https://arxiv.org/abs/1910.03875

[3] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks," [cs.CV], 2018. https://arxiv.org/abs/1611.07004v3

[4] Y. Lu, S. Wang, W. Zhao, and Y. Zhao, "WGAN-Based Robust Occluded Facial Expression Recognition," in *IEEE Access*, vol. 7, pp. 93594-93610, 2019. https://ieeexplore.ieee.org/document/8759893

[5] Y. Lu, X. Tao, N. Zeng, J. Du, and R. Shang, "Enhanced CNN Classification Capability for Small Rice Disease Datasets Using Progressive WGAN-GP: Algorithms and Applications," *Remote Sensing*, vol. 15, p. 1789, 2023. https://doi.org/10.3390/rs15071789.