

Assignment: Anime Face Generation using GANs

Introduction

Welcome to the assignment on Generative Adversarial Networks (GANs) for generating anime faces! This task will guide you through the process of building and training a GAN model on the Anime Face Dataset available on Kaggle.

Objective

Your objective is to implement a GAN model that can generate realistic anime faces. The model should be capable of learning the underlying patterns and characteristics of anime faces and generate new faces that resemble the dataset.

Dataset

- ***Dataset Name*:** Anime Face Dataset
- ***Dataset Source*:** [Anime Face Dataset on Kaggle]
(<https://www.kaggle.com/splcher/animefacedataset>)
- ***Description*:** This dataset contains thousands of anime face images, ideal for training GANs for image generation tasks.

Tasks

1. ***Data Preparation*:**

- Download the dataset from Kaggle.
- Preprocess the images (resize, normalize, augment if necessary).
- Prepare the dataset for training the GAN.

2. ***Model Architecture*:**

- Implement a GAN model architecture suitable for generating anime faces.
- Define the generator and discriminator networks.
- Choose appropriate activation functions, normalization layers, and other architectural decisions.

3. *Training the GAN*:

- Implement the training loop for the GAN.
- Train the model on the prepared dataset.
- Monitor and visualize the training process (loss curves, generated images).

4. *Evaluation*:

- Evaluate the quality of generated images using appropriate metrics (e.g., Inception Score, Frechet Inception Distance).
- Assess the diversity and quality of generated anime faces.

5. *Documentation and Reporting*:

- Document your code with clear comments and explanations.
- Prepare a report summarizing your approach, challenges faced, and results.
- Include visualizations of generated images and evaluation metrics.

Deliverables

- Python script for the complete GAN pipeline.
- Documentation (README.md) detailing:
 - Instructions for running the code.
 - Description of the GAN architecture.
 - Explanation of the training process.
 - Evaluation results and analysis.

Resources

- Required libraries: TensorFlow, PyTorch, or any preferred deep learning framework.
- Kaggle API (for downloading the dataset if using Colab or similar environment).

Submission

Submit the following:

- Complete Python script (gan_anime_faces.py).
- README.md file with documentation.
- Any additional files or resources used.

Additional Notes

- Ensure your code is well-structured and follows best practices in deep learning.
- Experiment with different hyperparameters and model architectures to improve results.
- Feel free to include additional features or enhancements beyond the basic requirements.