NYU

HPML Final Project Presentation

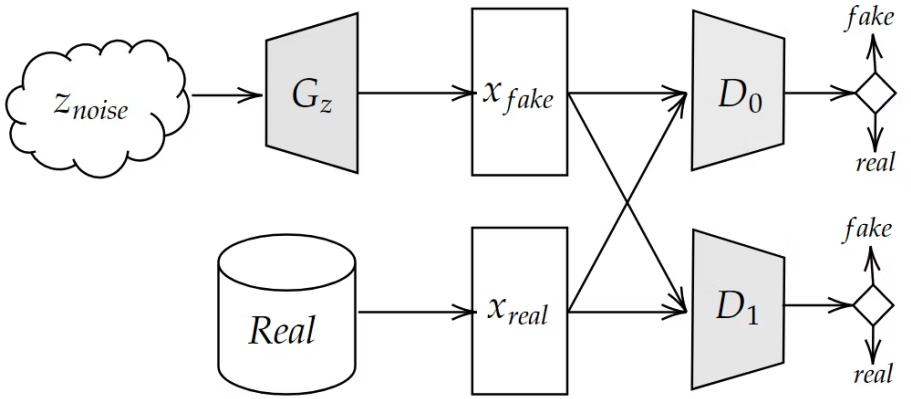
# Double the Discrimination: Enhancing DCGAN with Dual Discriminators

Ishaan Pitty, Savinay Shukla

ip2244, ss16924

# Executive Summary

- **Goal**: Faster convergence for GANs.
- **Discoveries**: Training instabilities with DCGAN and mode collapse. Requirement for better evaluation metrics.
- **Plan of Attack**: Monitoring sensitivities to convergence by introducing another discriminator. Faster convergence is possible. More possibilities of pipelining.
- **Improved evaluation metrics**: Fréchet Inception Distance (FID) & Inception Score (IS) for better image quality/diversity assessment.
- **Distributed Analysis**: Scalability of our approach across multiple GPUs. Can we reduce compute costs with early stopping?




Our proposed training pipeline with two discriminators.


# Technical Challenges

**For Single-Node Benchmarking:**

- **Generalizing** against a diverse set of datasets.
- Implementations of **FID and IS scores**. Inception networks add memory overhead. Slow evaluations.

**For Distributed Benchmarking:**

- **Consistency** across architecture.
- **Communication overhead** by adding more backward passes.
- **IS score benchmarking** across iterations. Incompatibility with GLOO and NCCL backends.
- **GLOO and NCCL** backends. Slow warmups?

# Benchmarking Metrics

**Datasets**: CIFAR-10, CIFAR-100, MNIST, SVHN

**GPUs** : NVIDIA RTX 3060 Ti, NVIDIA RTX 3070 Ti, NVIDIA Tesla V100

**OS**: Linux and Windows.

**Framework**: Pytorch and Pytorch DDP

# Proposed Methodology

**Step 1 - Benchmark!**

- **Benchmarking** DCGAN across multiple datasets.
  - **Datasets**: **CIFAR 10, MNIST, SVHN**, and others, across **multiple batch sizes**. **Across different GPUs**.
  - We used this as the benchmark for our improvised implementation.

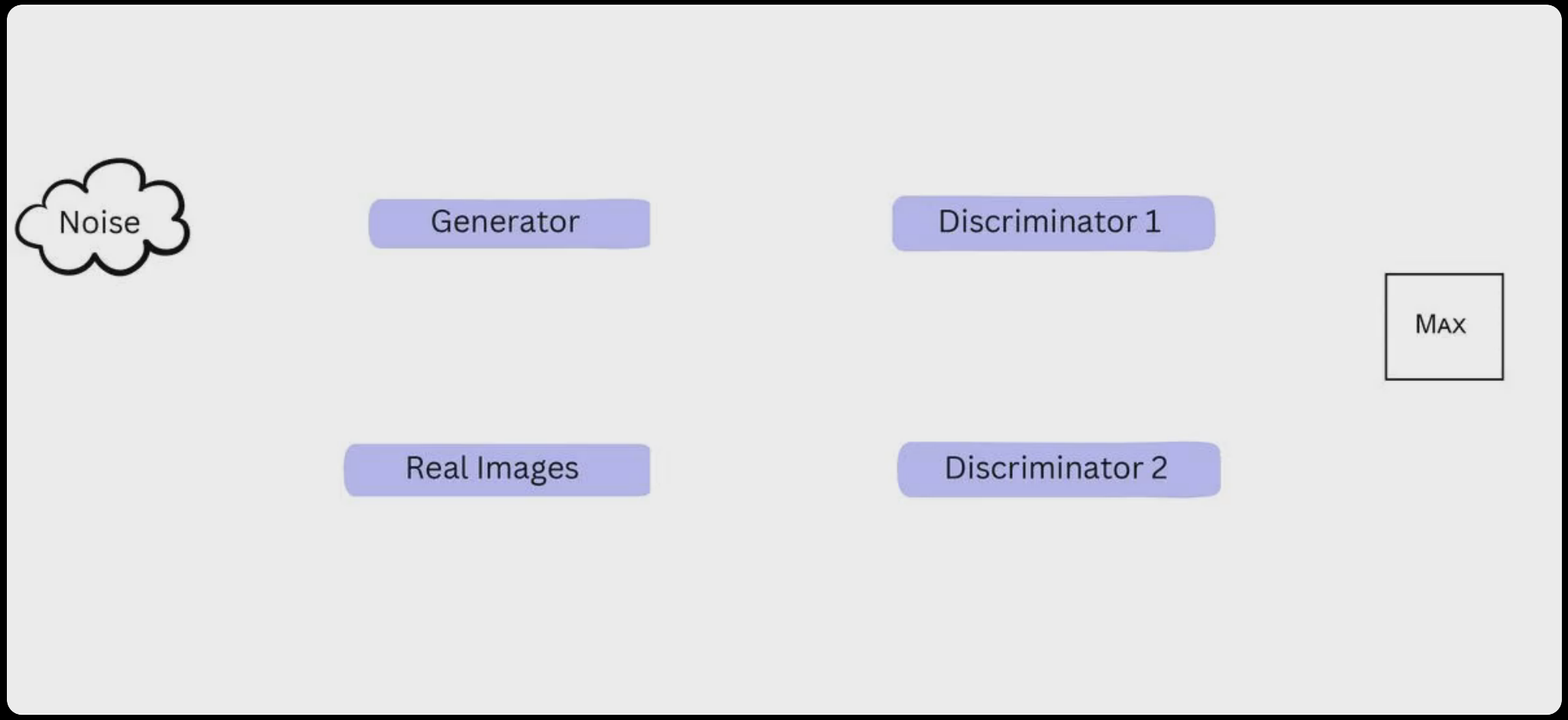
**Step 2 - Improvise!**

- Modified DCGAN by adding another **diversity discriminator**.
  - The generator has to **fool two discriminators instead of one**.
  - Discriminators update themselves based on the combined losses. Adds self-regularization.
  - The generator updates itself based on the **max loss** produced by either of the discriminators. Faster convergence.

**Step 3 - Extend and analyze!**

- Wrapping up our model using PyTorch's Distributed Data-Parallel module, and observing scalability.

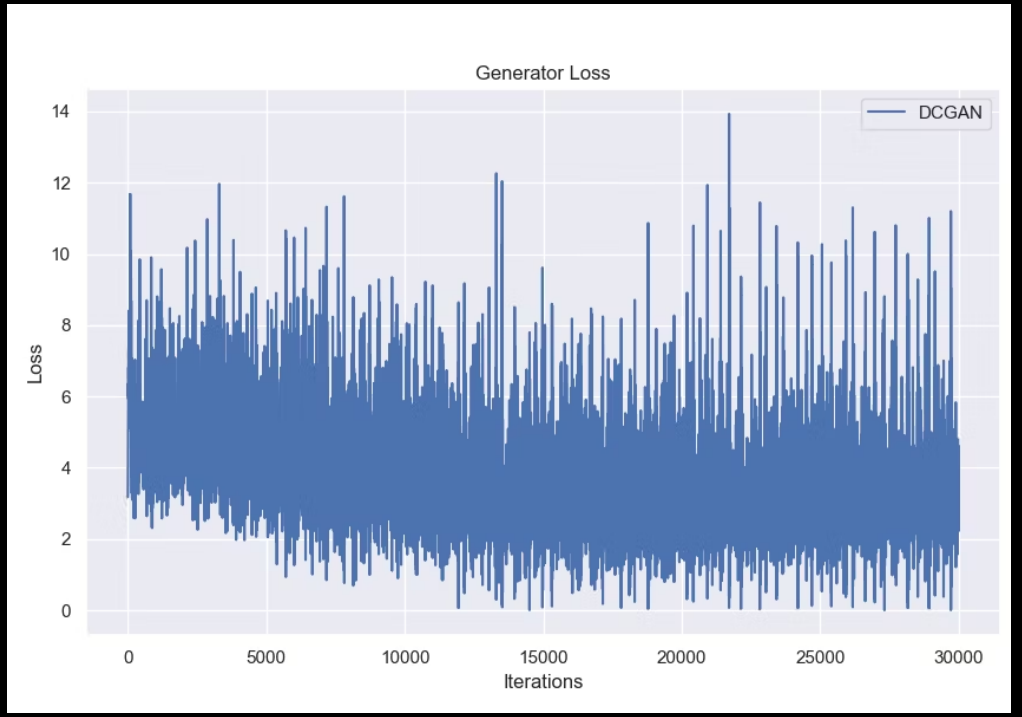
# Improvised training pipeline with an additional discriminator



# Evaluation Metrics?

Better evaluation metrics are required for the proper evaluation of generated data/images. Generator or discriminator loss is not always an accurate metric for quality.

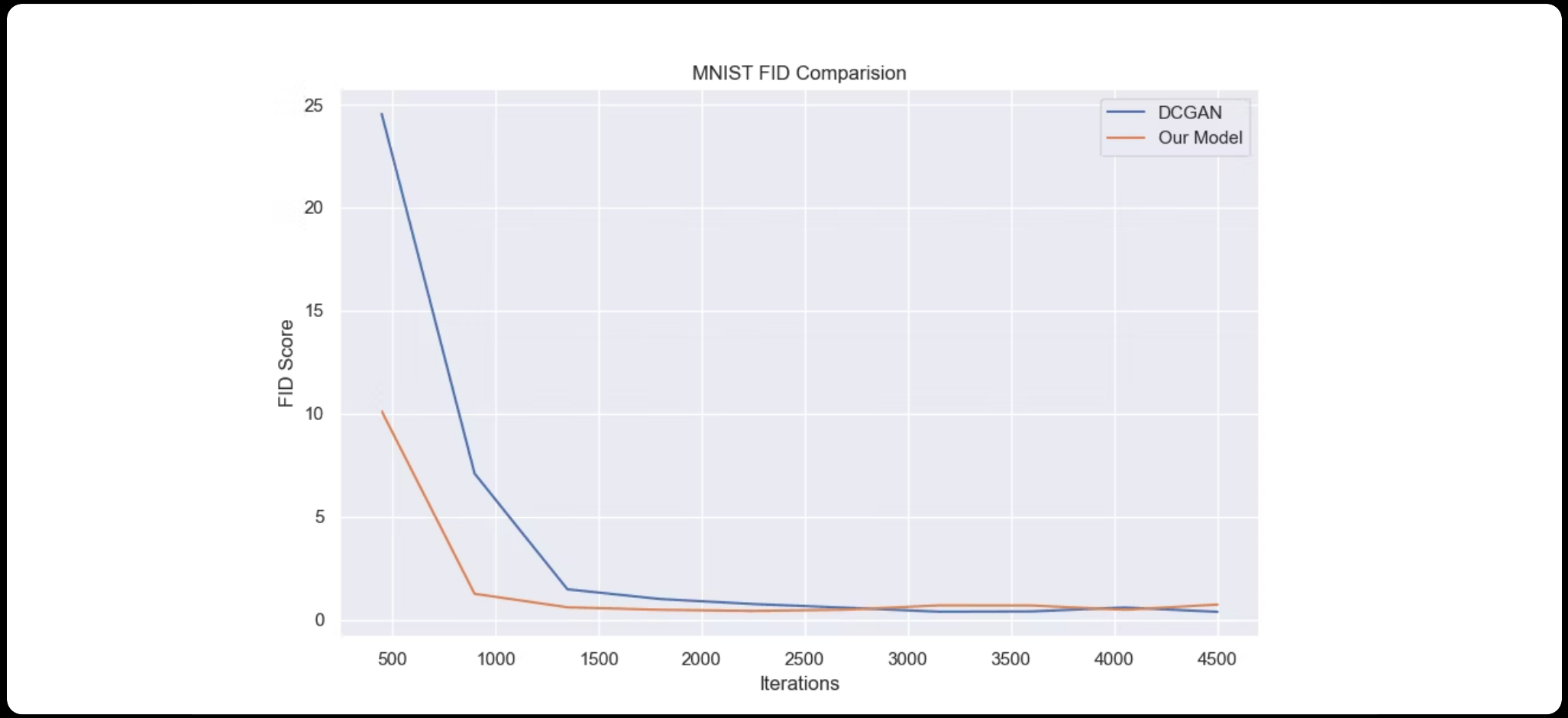
Better options? - ***Inception Score and FID Scores***.



NYU

# Performance Evaluation Results

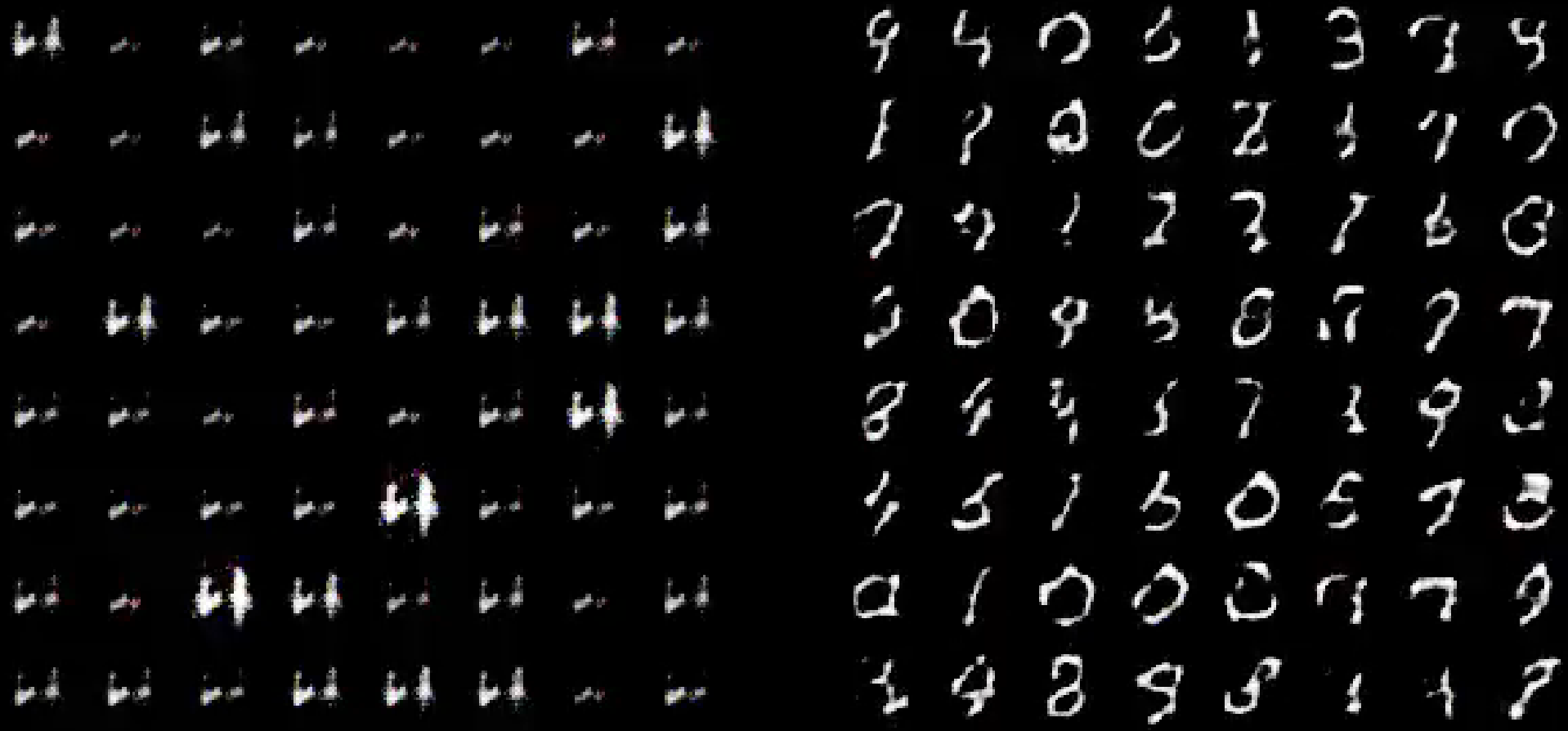
Using the official implementions of Inception Scores and FID scores.




# FID Score Comparision - MNIST (50000 images)

# DCGAN vs Modified DCGAN - MNIST

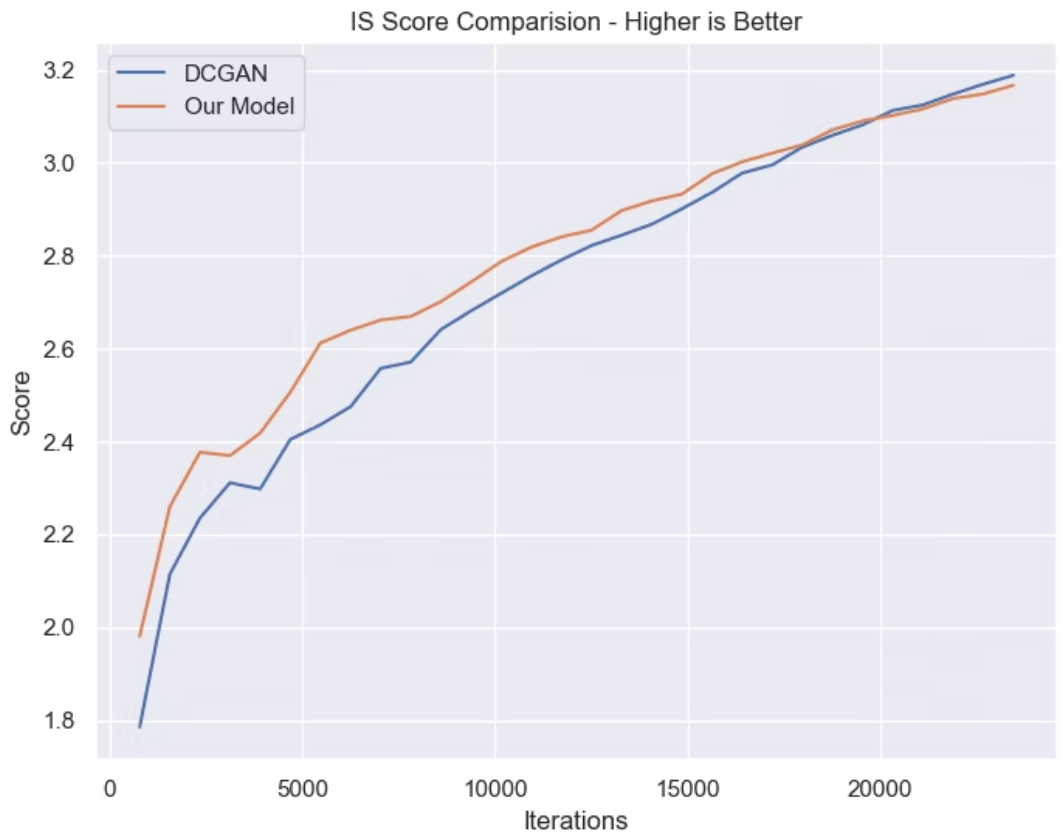
# DCGAN vs Modified DCGAN - MNIST



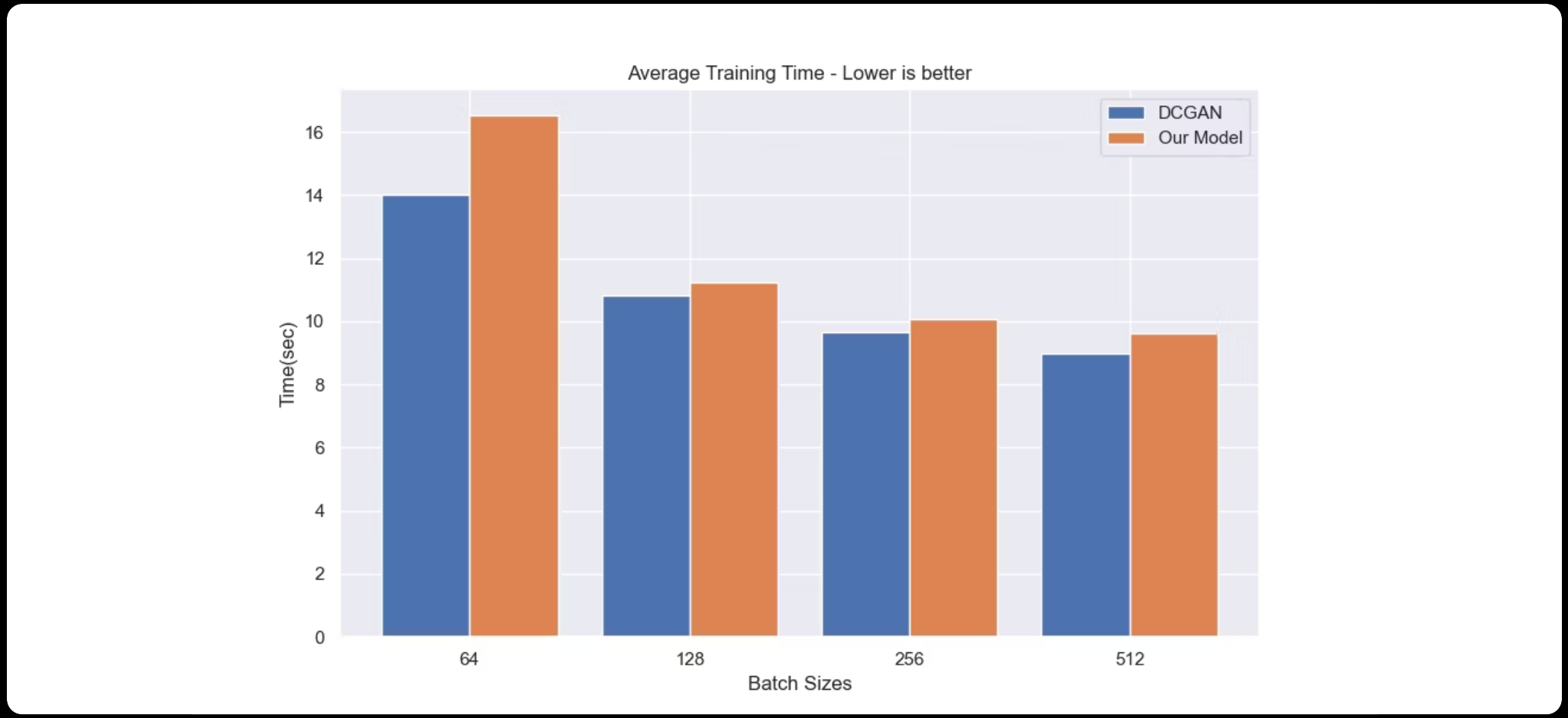




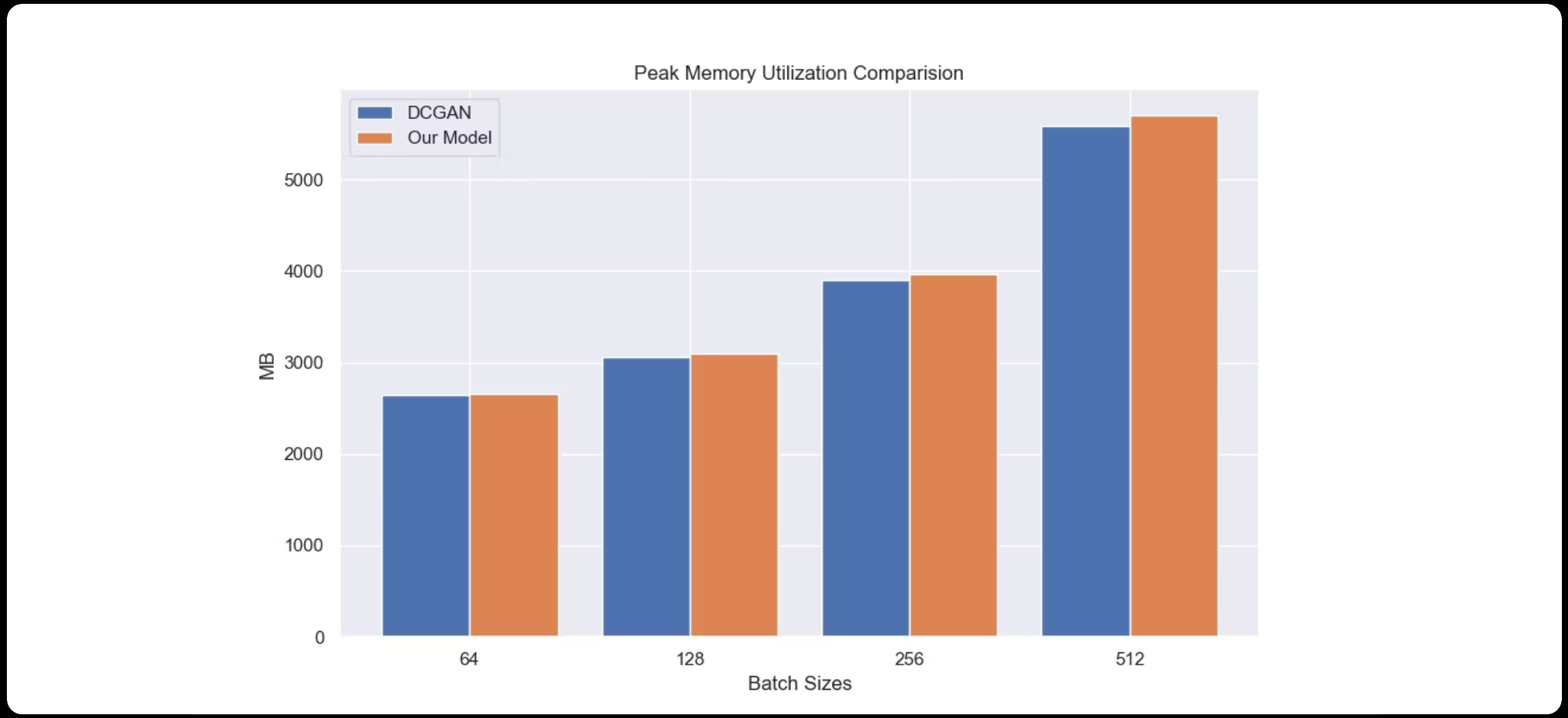
# FID Score Comparision - CIFAR 10



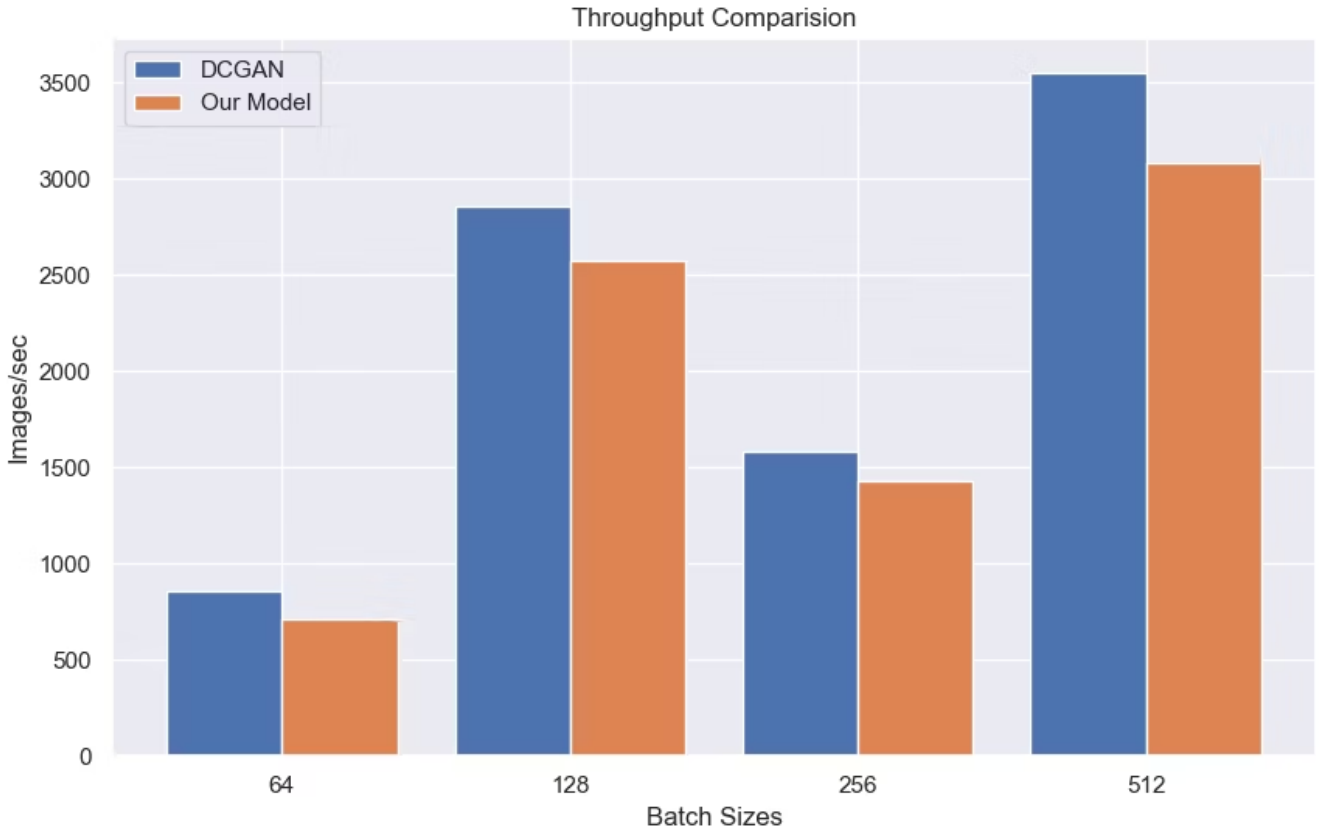

# IS Score Comparision - DCGAN vs Modified DCGAN




# Training Time Comparision.

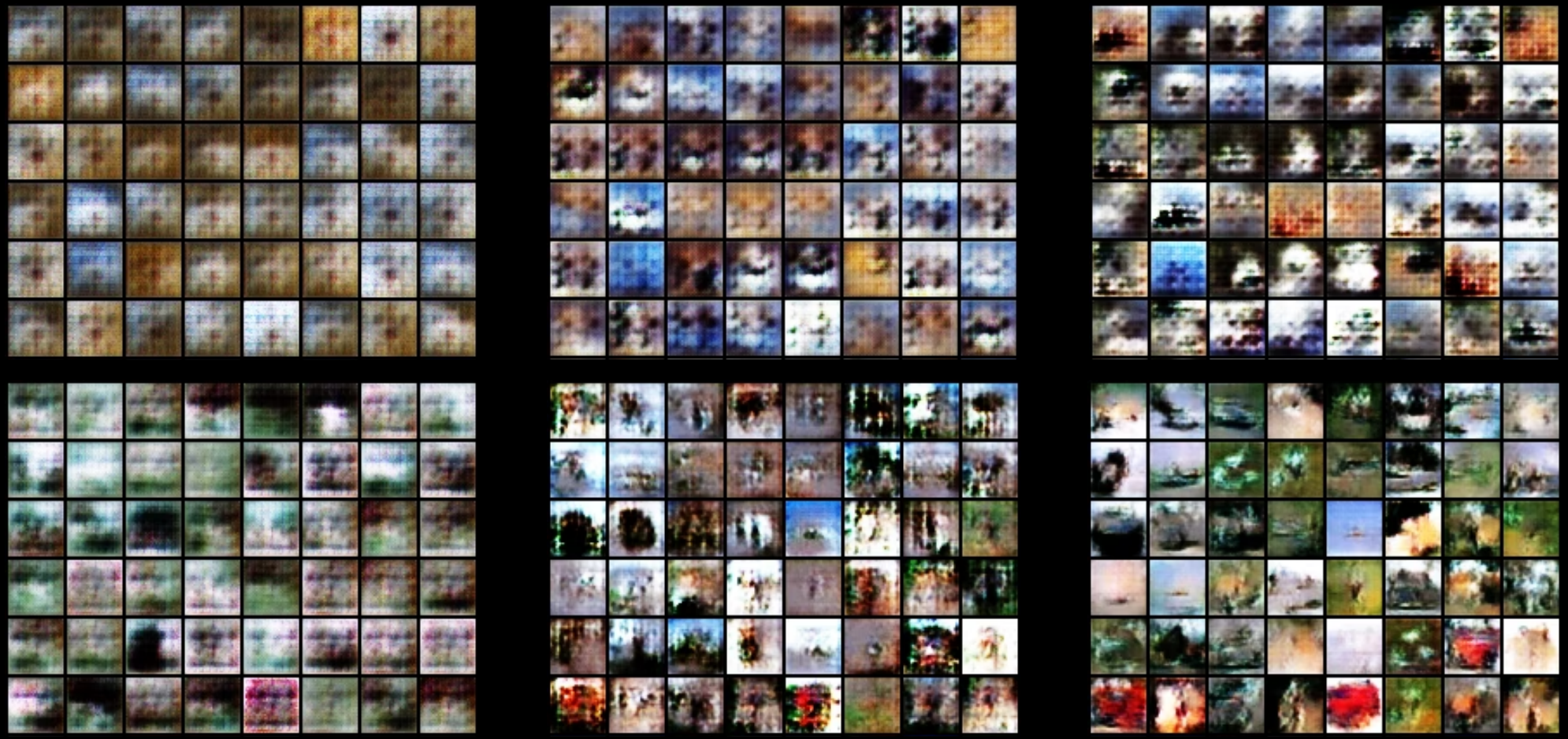



# Peak Memory Utilization : Tesla V100



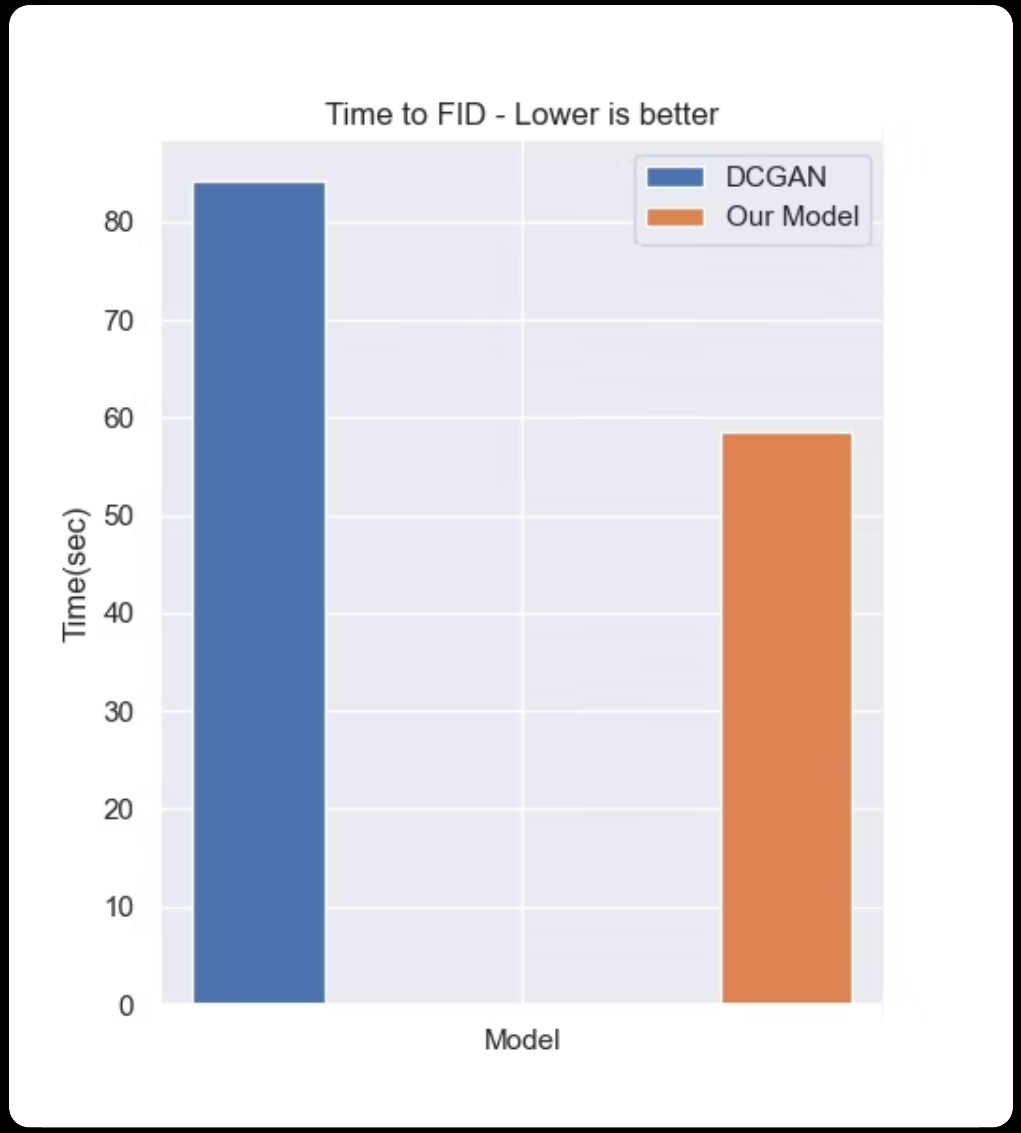

# Throughput Comparision : Tesla V100

# DCGAN vs Our Pipeline - CIFAR10



# Speedup - 31.2%

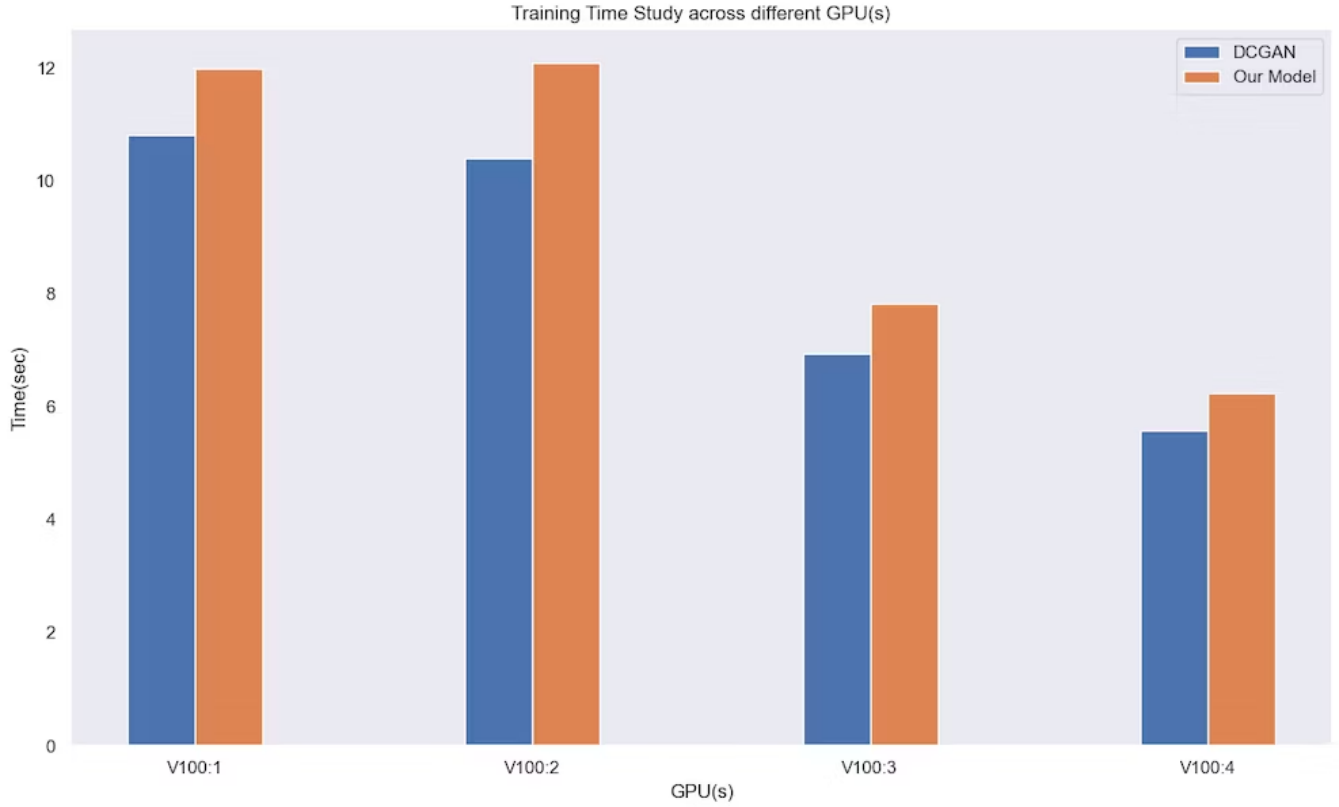
Time to achieve optimal FID score - CIFAR 10



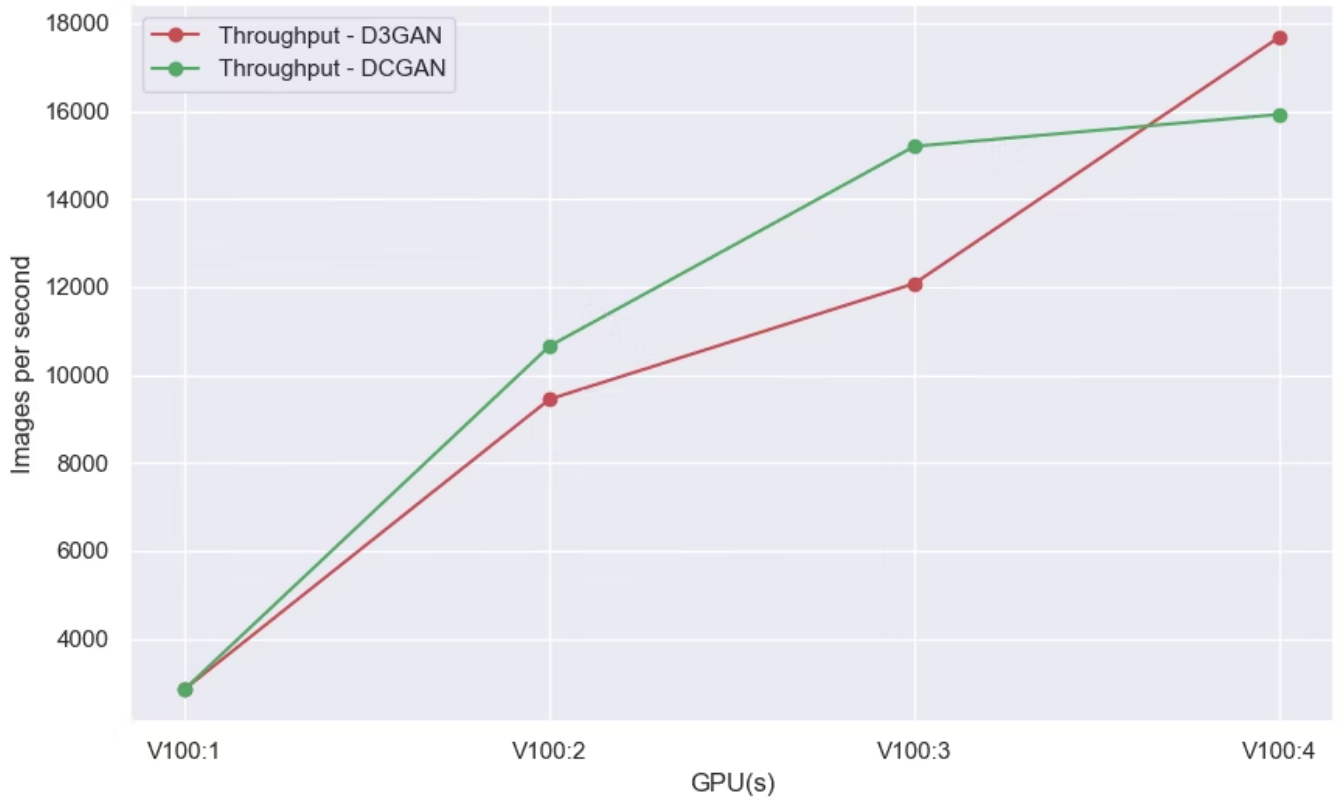
NYU

# Distributed Double Discrimination

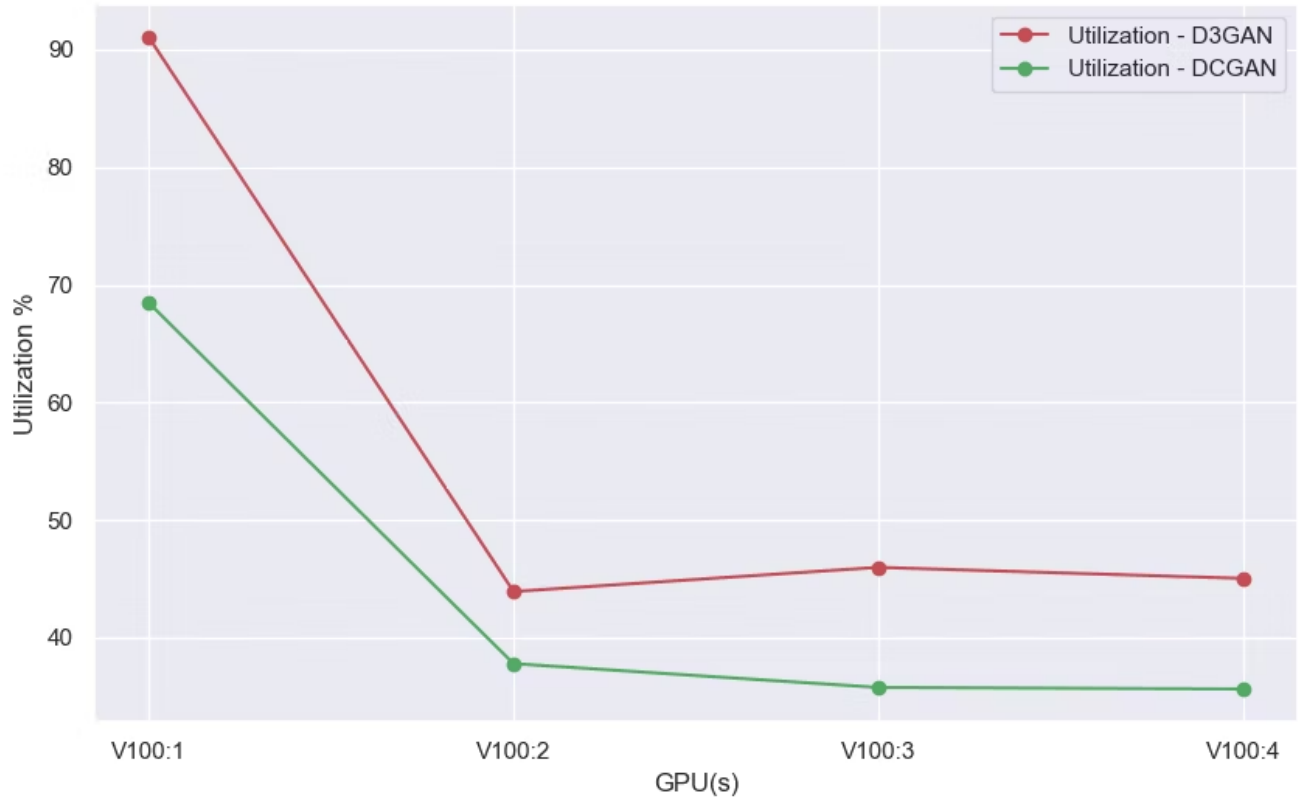
We wrapped our training pipeline using PyTorch's DDP module to observe scalability and limitations.




Training Time (per epoch) across multiple GPUs.




# Throughput Comparision




# GPU Utilization Comparision

NYU

# Observations & Conclusions

- Our experimental results show that the addition of another discriminator improves the quality of generated images and its convergence is must faster.
- Our pipeline holds true across the majority of datasets that are widely used for generative modeling.
- Even though training time increases because of an additional discriminator, the overall time to converge is much lesser than DCGAN.
- We strongly believe that this pipeline could be generalized for multiple discriminators in the network.
- Stronger and faster convergence could be achieved with more sophisticated and deeper networks.

NYU

# GitHub Repo

https://github.com/SavinayShukla/Distributed-Dual-Discriminator-GANs