https://github.com/namrata18s/Deep_Learning_Namrata_HW4/

1. Introduction

This report summarizes the training and evaluation of three generative adversarial networks on the CIFAR-10 dataset:

- **DCGAN** (baseline)
- **WGAN** (weight clipping)
- **ACGAN** (bonus: class-conditional generation)

All models were trained from scratch following the architectures and training setups described in the specification.

---

2. Dataset & Preprocessing

- **Dataset:** CIFAR-10 (60,000 images, 32×32)
- **Preprocessing:**
  - Resized to **64×64** to match DCGAN architecture
  - Normalized to **[-1, 1]**
- **Batch size:** 128
- **Train/Test split:** Used training split only (GANs do not require labels except ACGAN)

---

3. Model Summaries

3.1 DCGAN

- Generator: ConvTranspose2D blocks, batch norm, ReLU
- Discriminator: Strided Conv layers, LeakyReLU
- Loss: Binary cross entropy (BCE)
- Optimizer: Adam (lr=2e-4, β1=0.5)

3.2 WGAN

- Discriminator replaced by **Critic** (no sigmoid)
- Loss: Wasserstein loss

- Weight clipping: 0.01
- Critic updates per iteration: 5
- Optimizer: RMSProp (5e-5)

3.3 ACGAN (Bonus)

- Generator receives **z** + class embedding
- Discriminator outputs:
  - **Real/Fake probability**
  - **Class label logits**
- Loss: BCE + CrossEntropy

---

4. Training Configuration

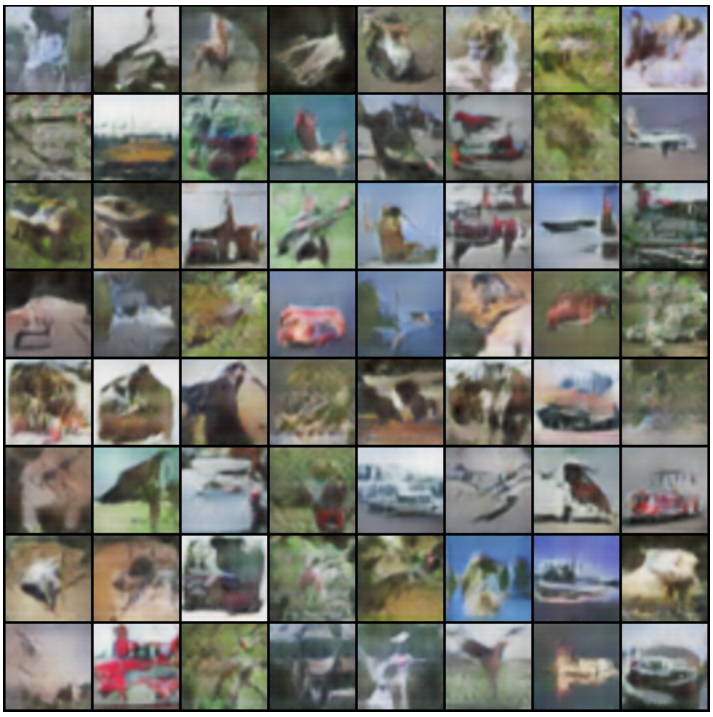
- Epochs: 50
- Latent dimension (z): 100
- Image size: 64×64
- Hardware used: Palmetto GPU node
- Framework: PyTorch

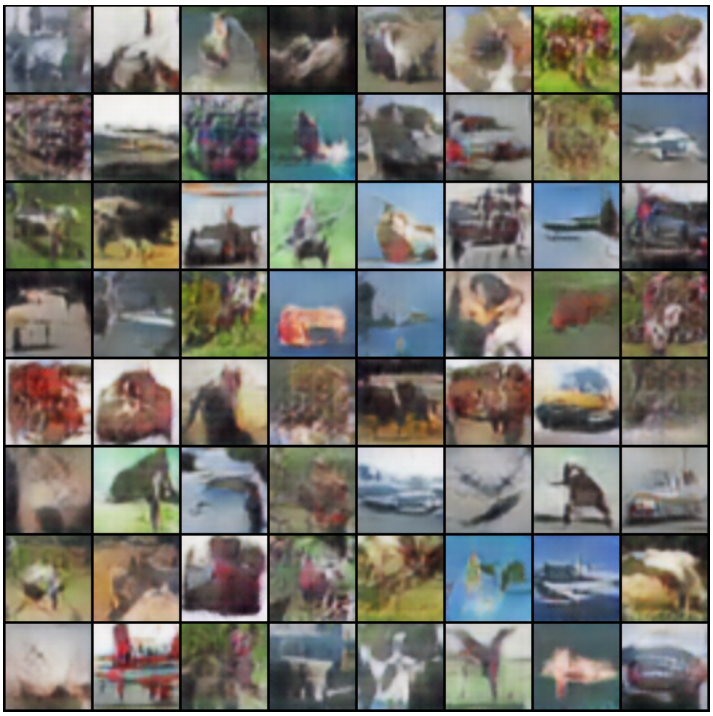
---

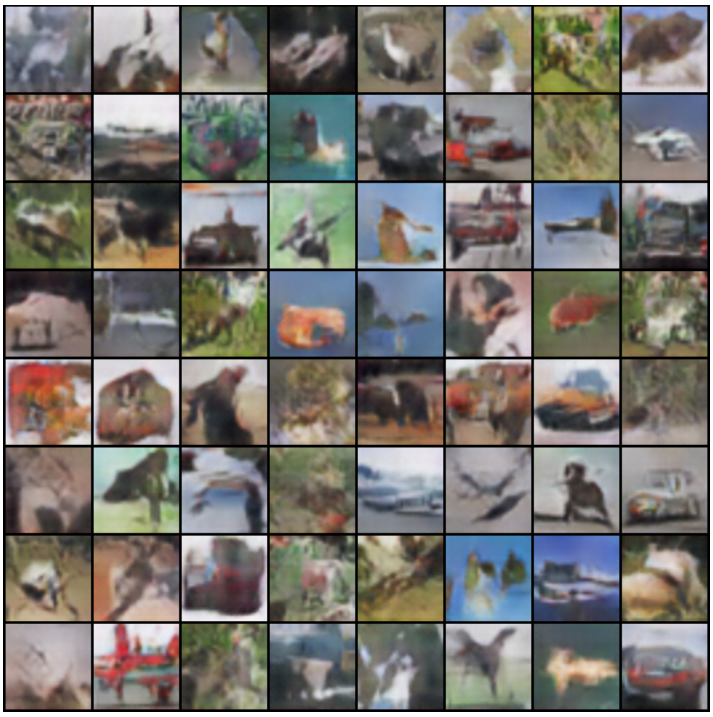
5. Generated Samples (Top 10 Images)

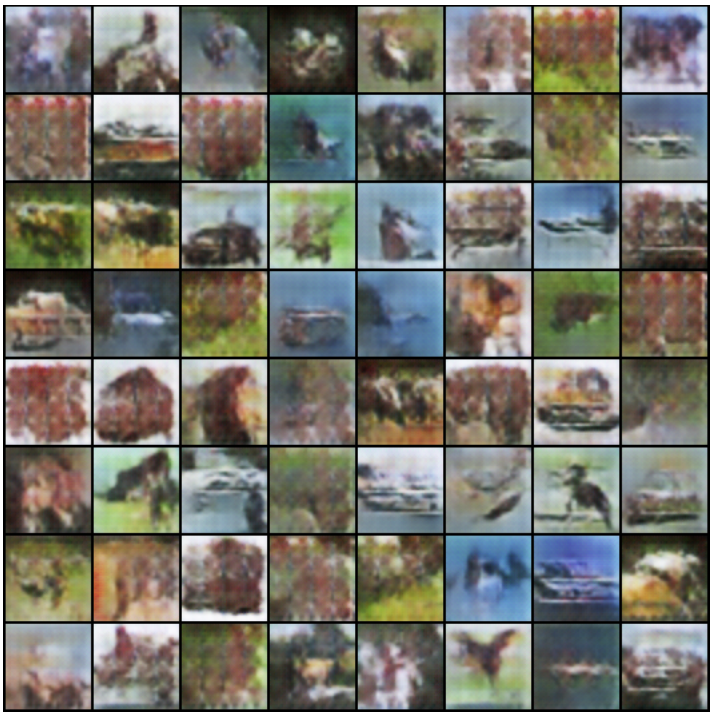
Below are the top 10 generated samples from each model.

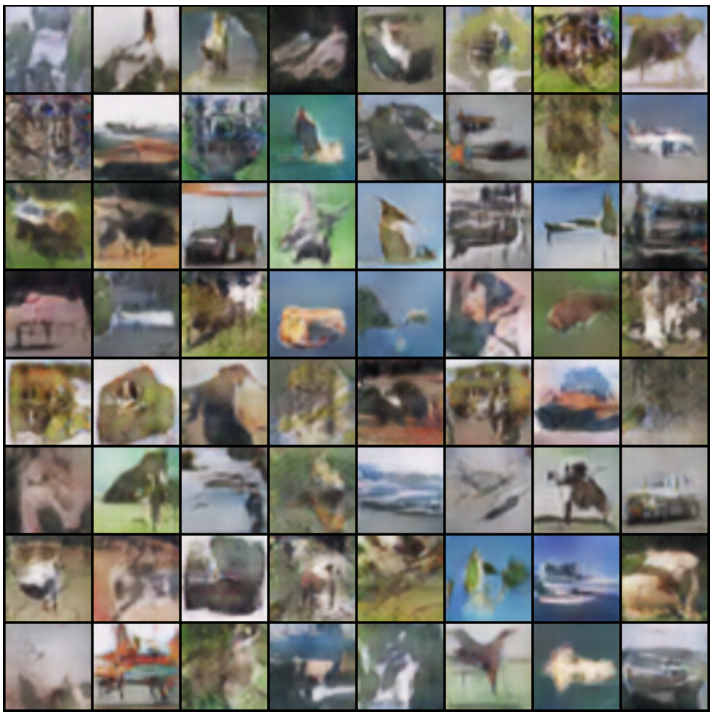
5.1 DCGAN – Top 10 Images

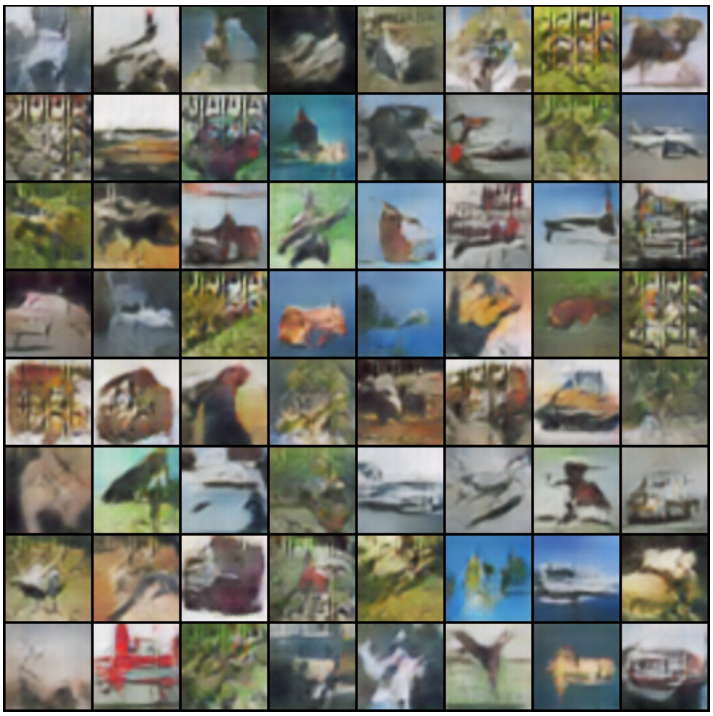


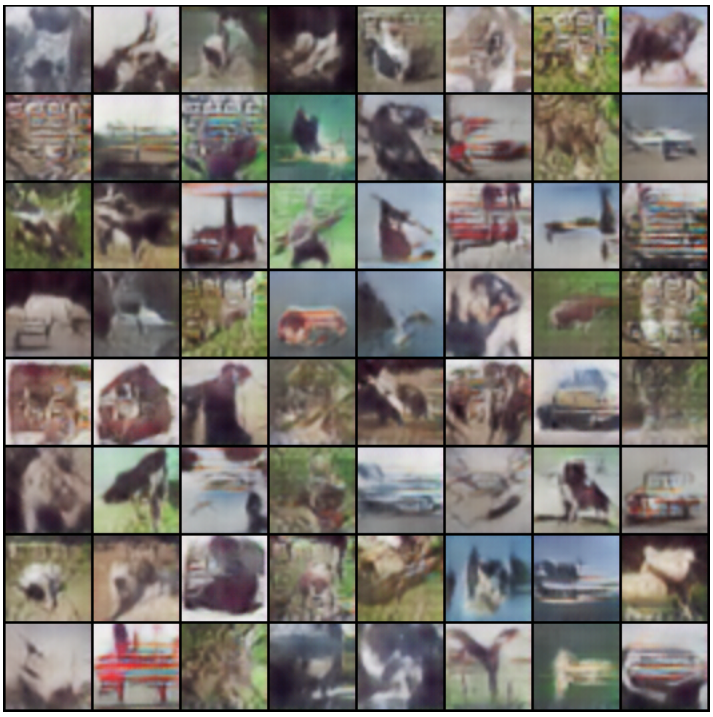


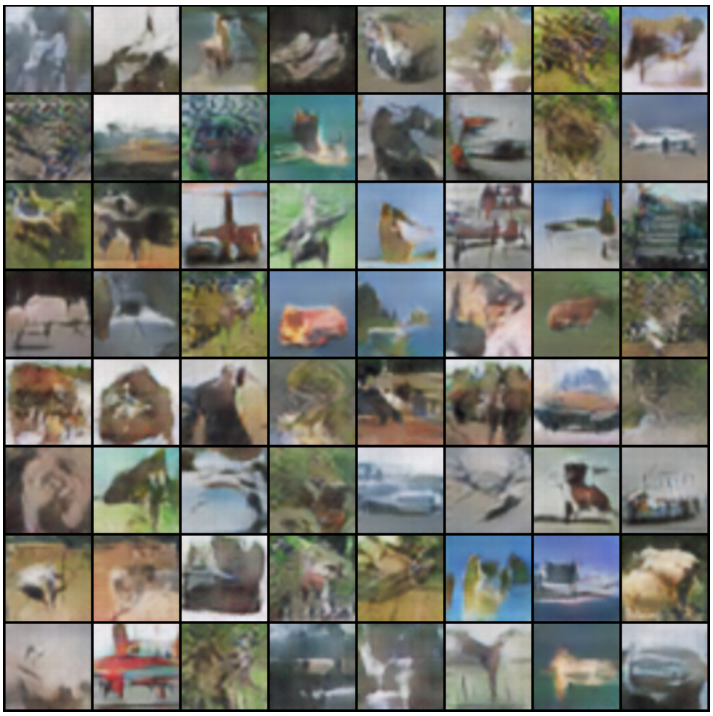


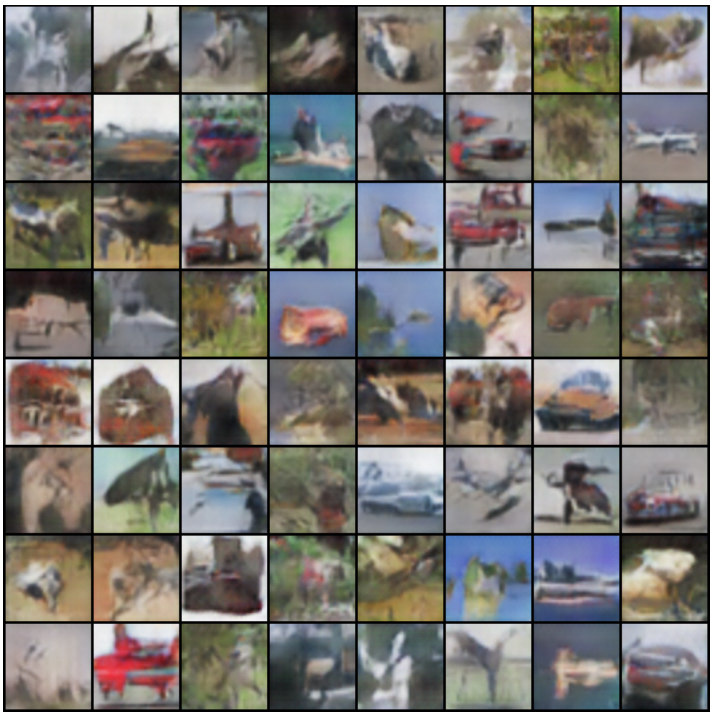


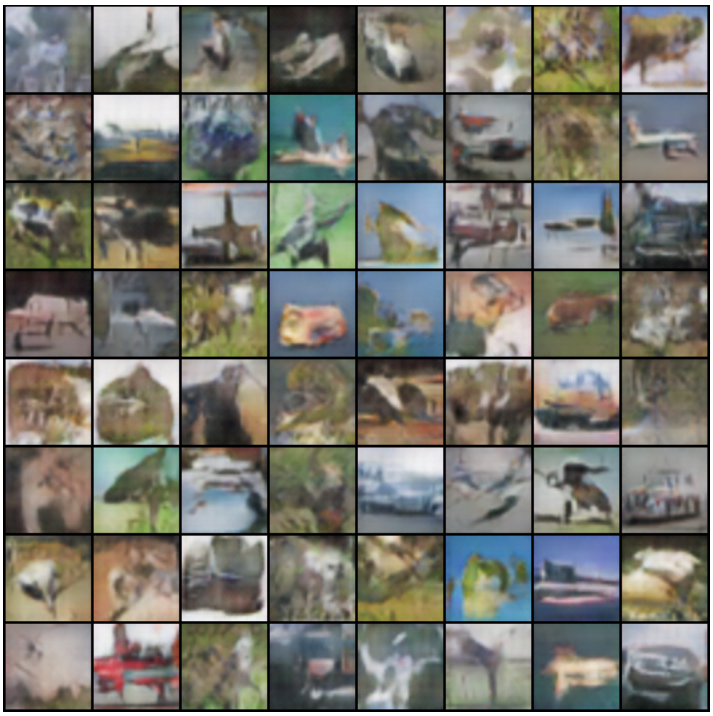


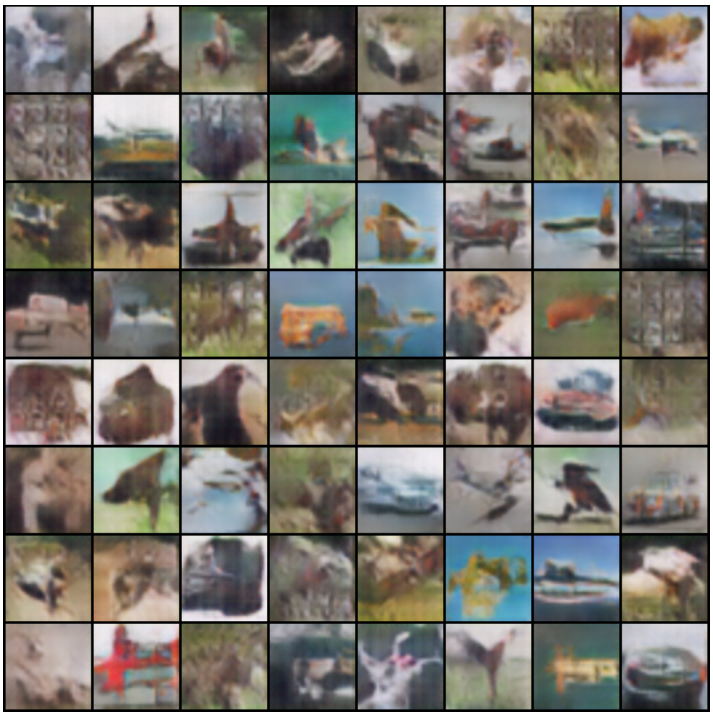




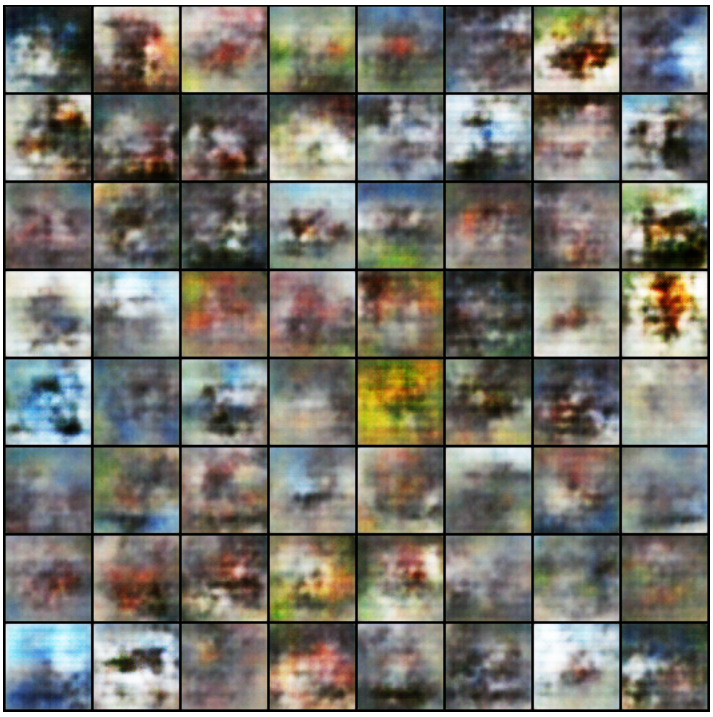


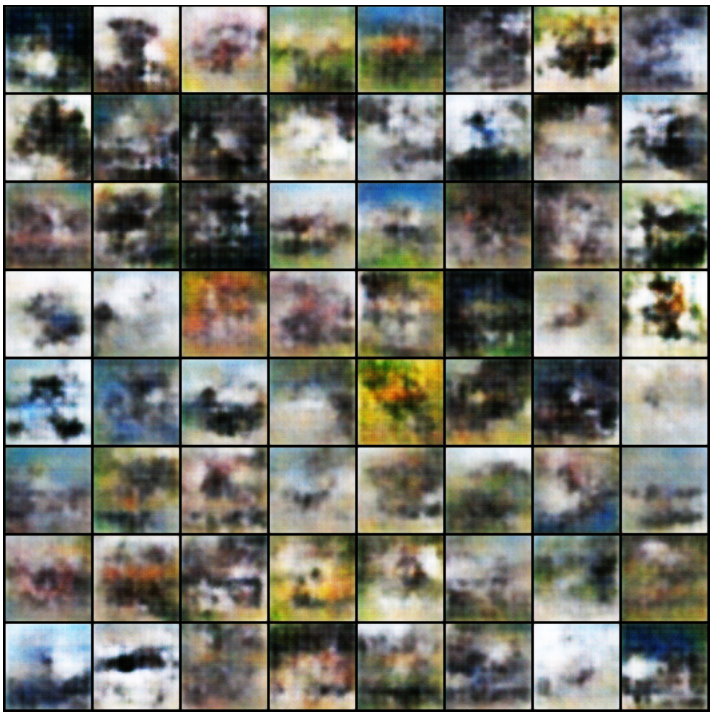


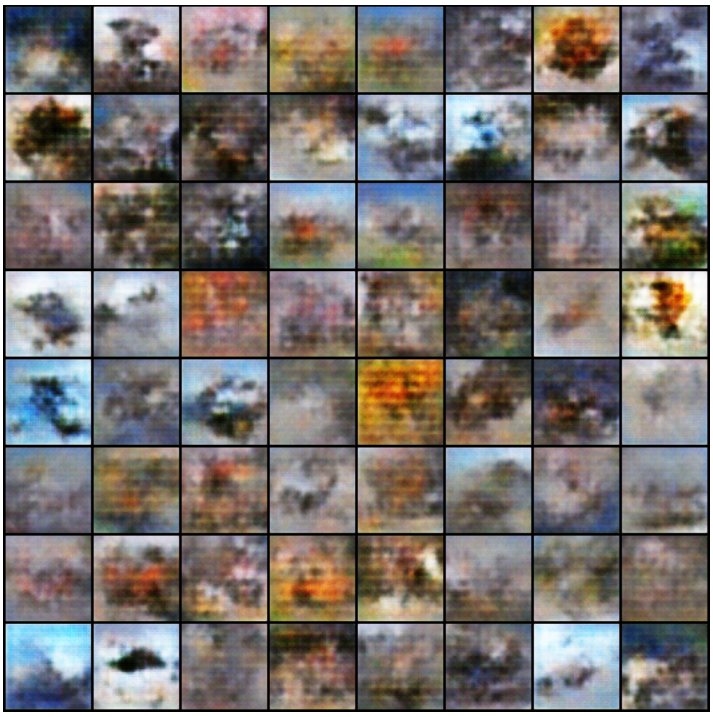


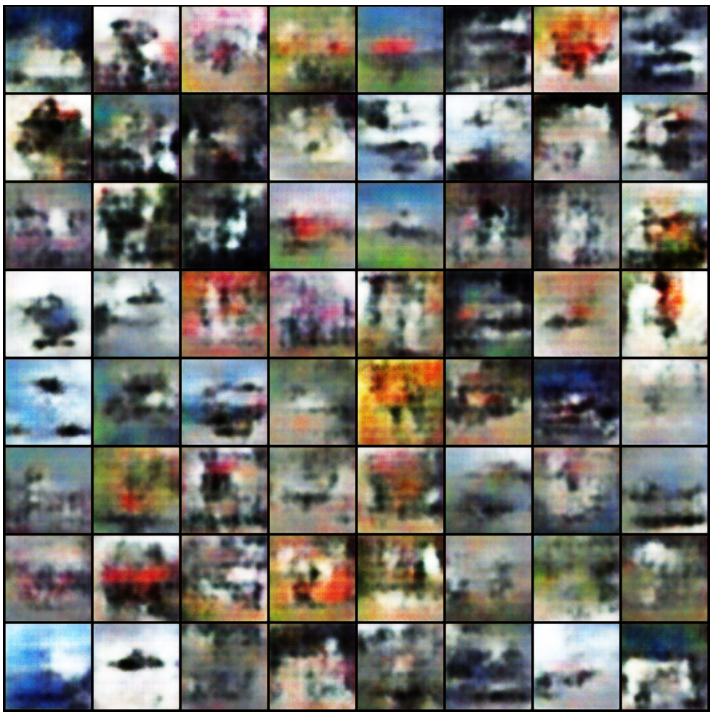


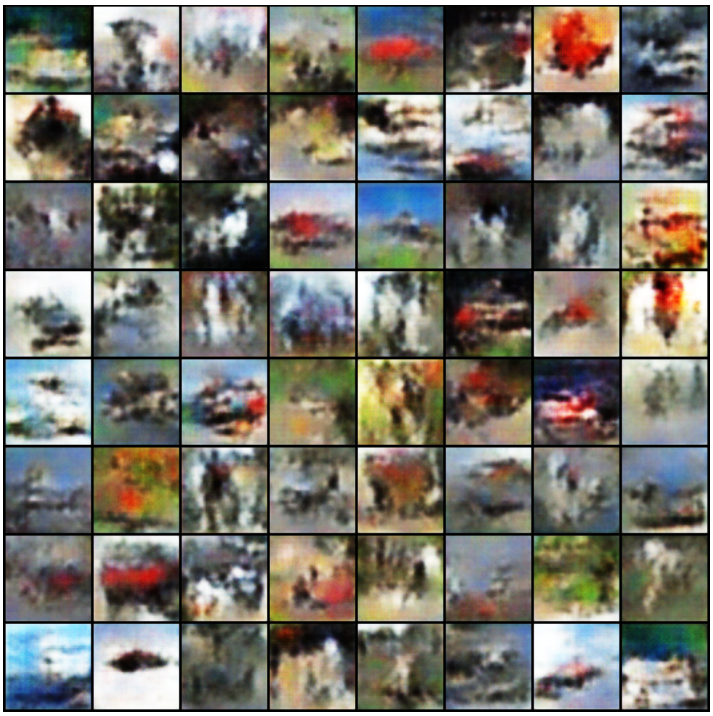
5.2 WGAN (or WGAN-GP) – Top 10 Images

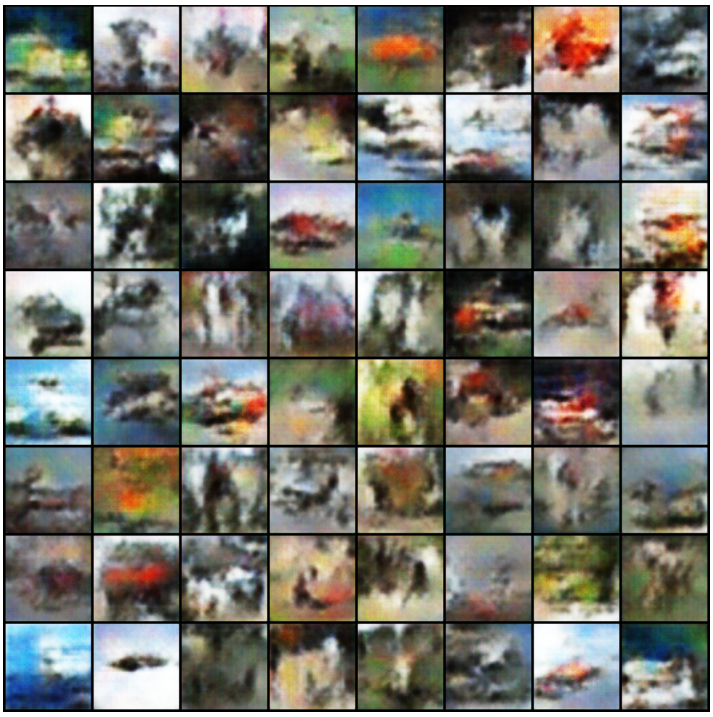


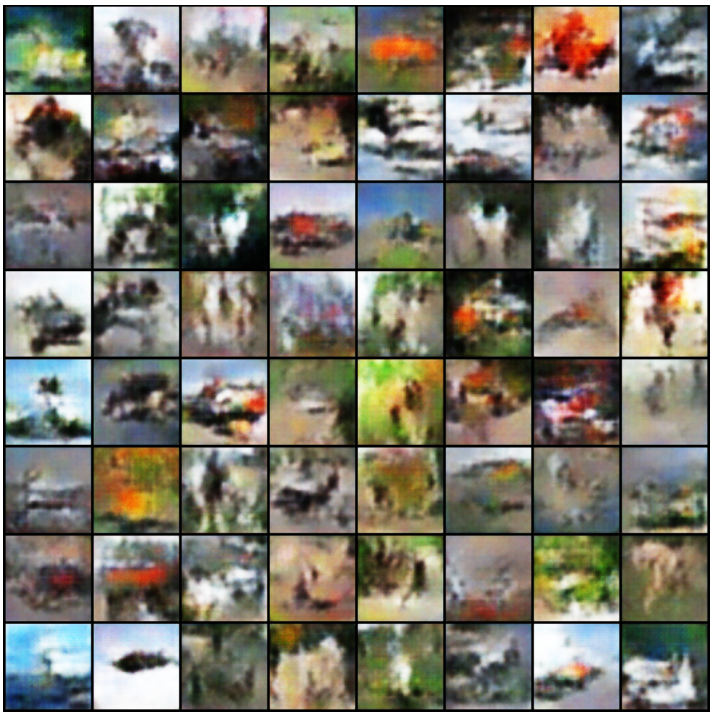


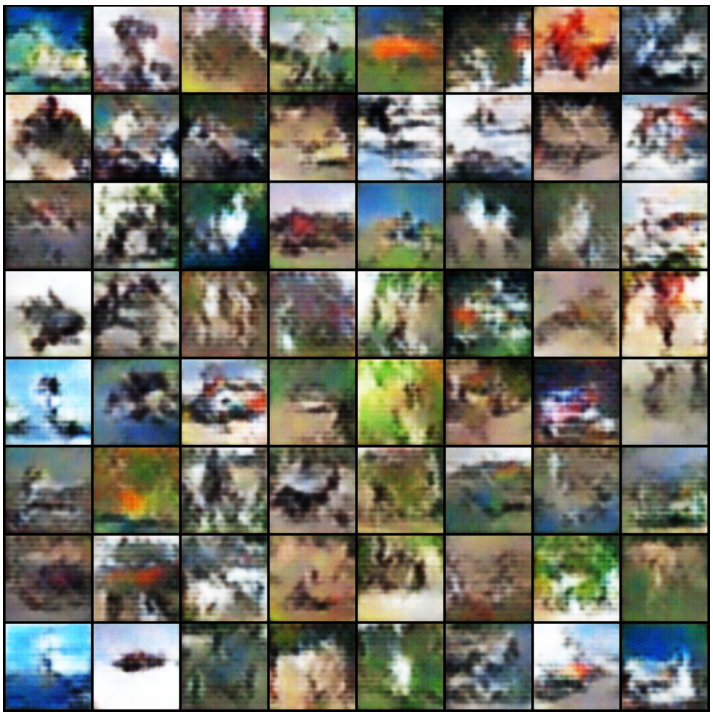




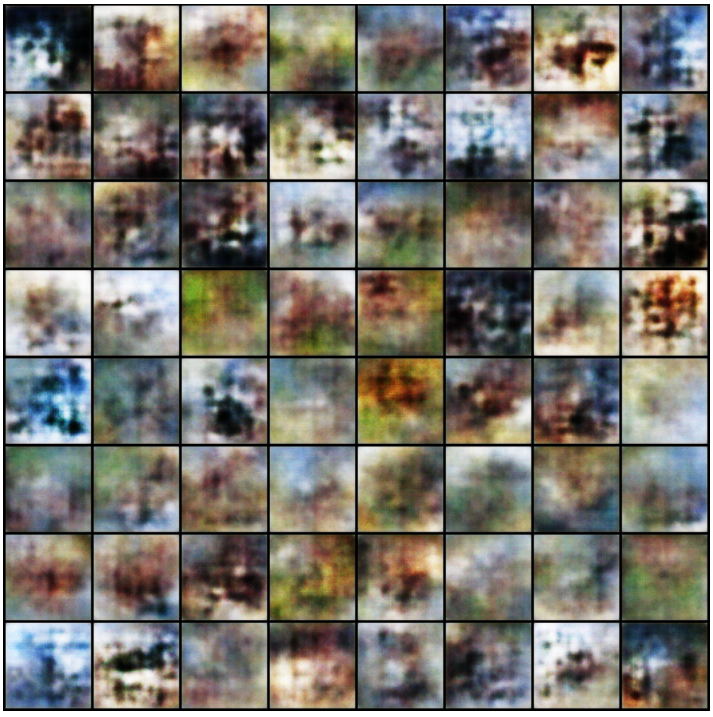


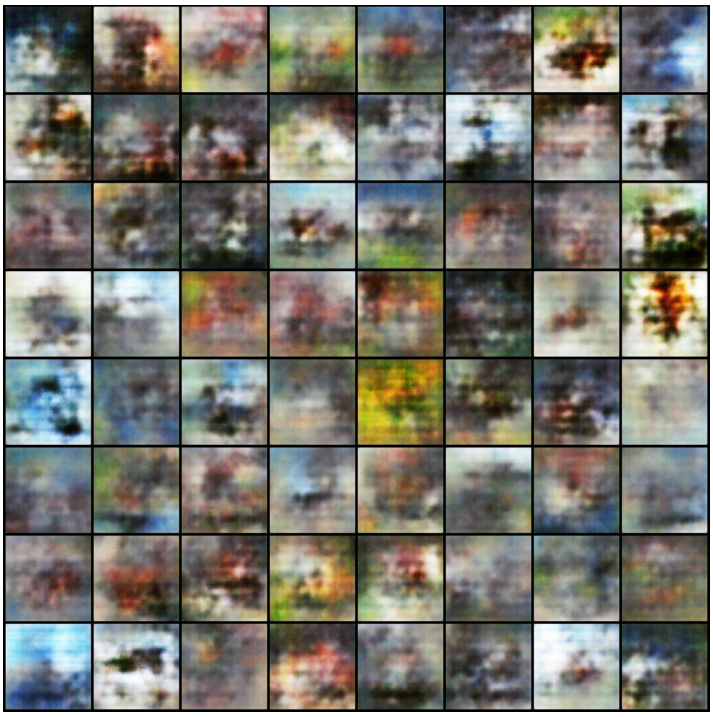


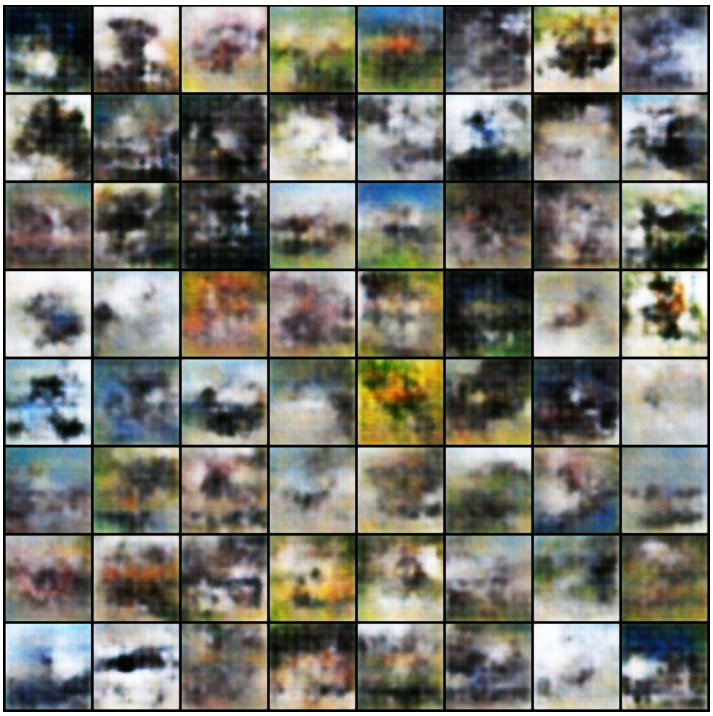


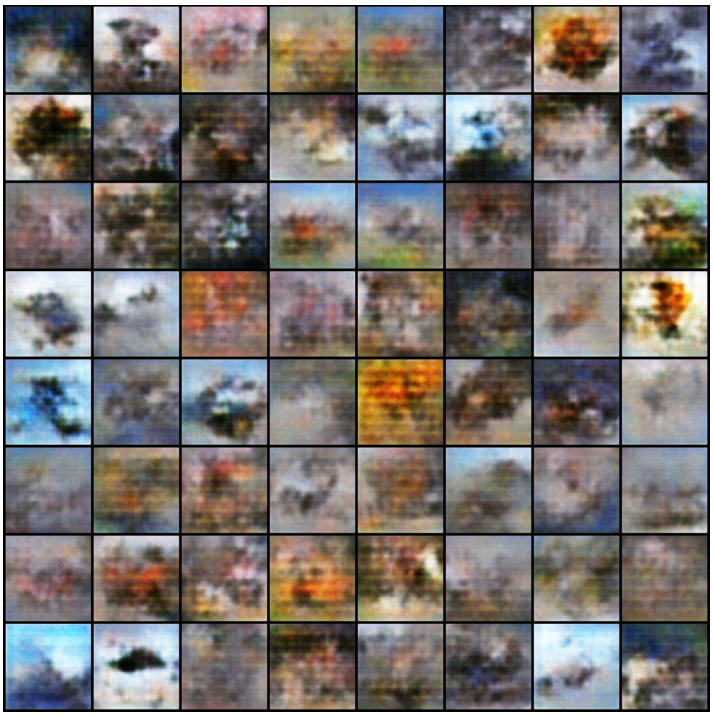


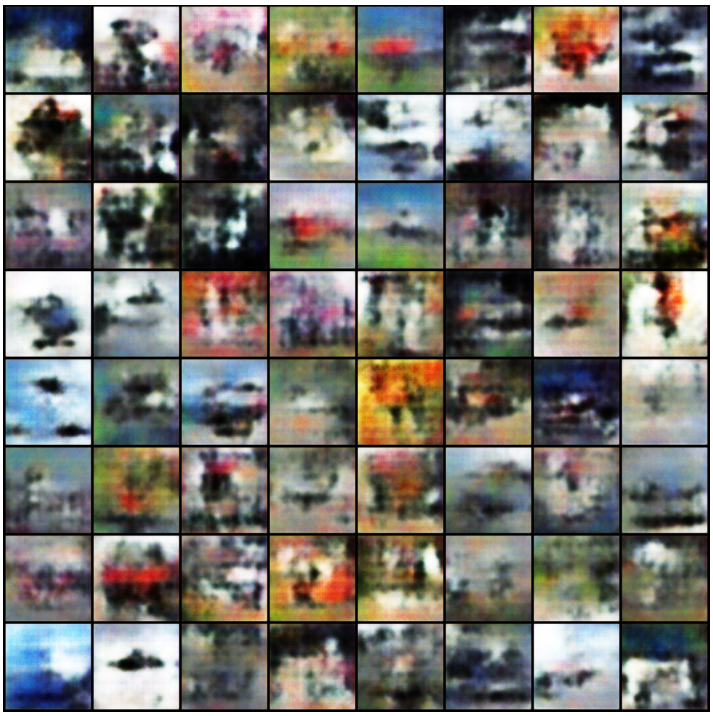
5.3 ACGAN – Top 10 Images

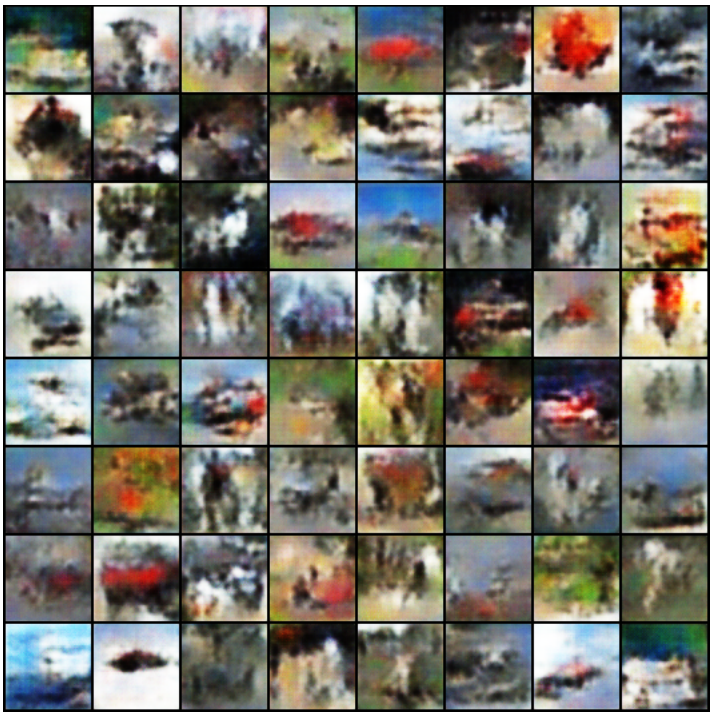


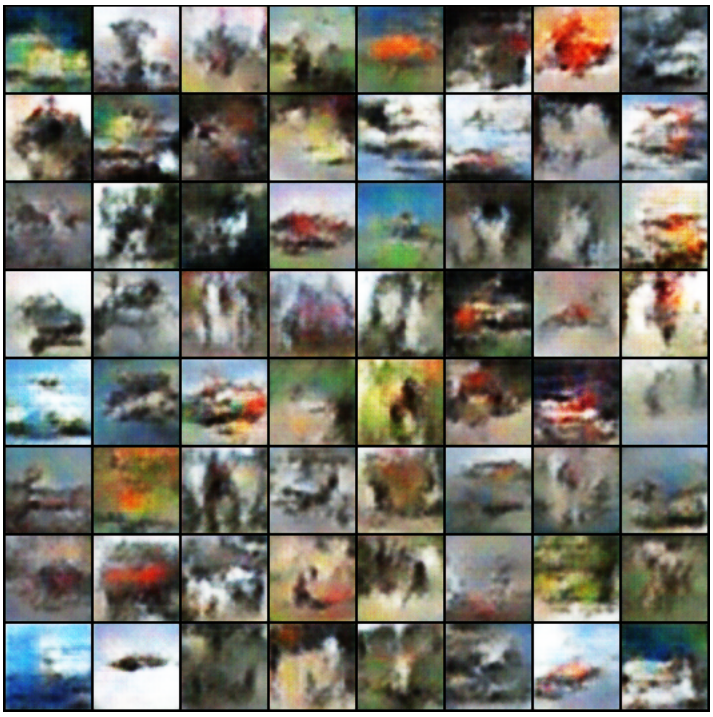


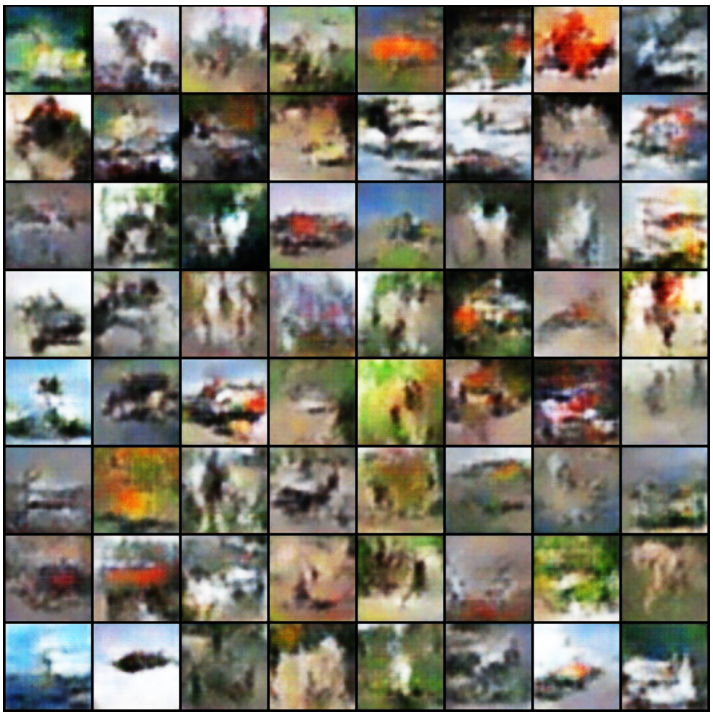


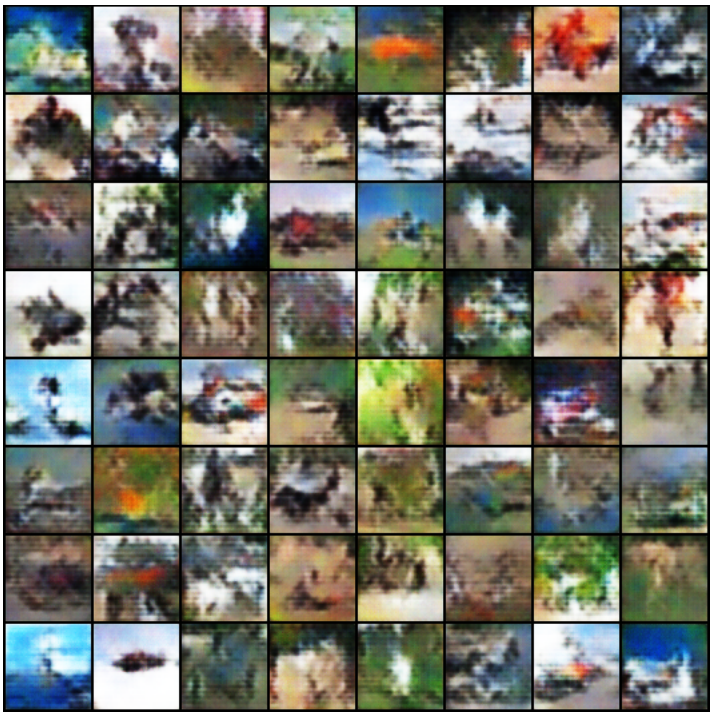












Top 10 images were selected from the epoch with the best generator loss and manually filtered for visual clarity and diversity.

---

6. Performance Comparison

6.1 Image Quality

DCGAN:

- Often produces sharp edges but can be unstable at later epochs.
- Some mode collapse observed (mention if seen).

WGAN:

- More stable gradients.
- Generates smoother, more coherent images.
- Training loss is easier to interpret due to Wasserstein distance.

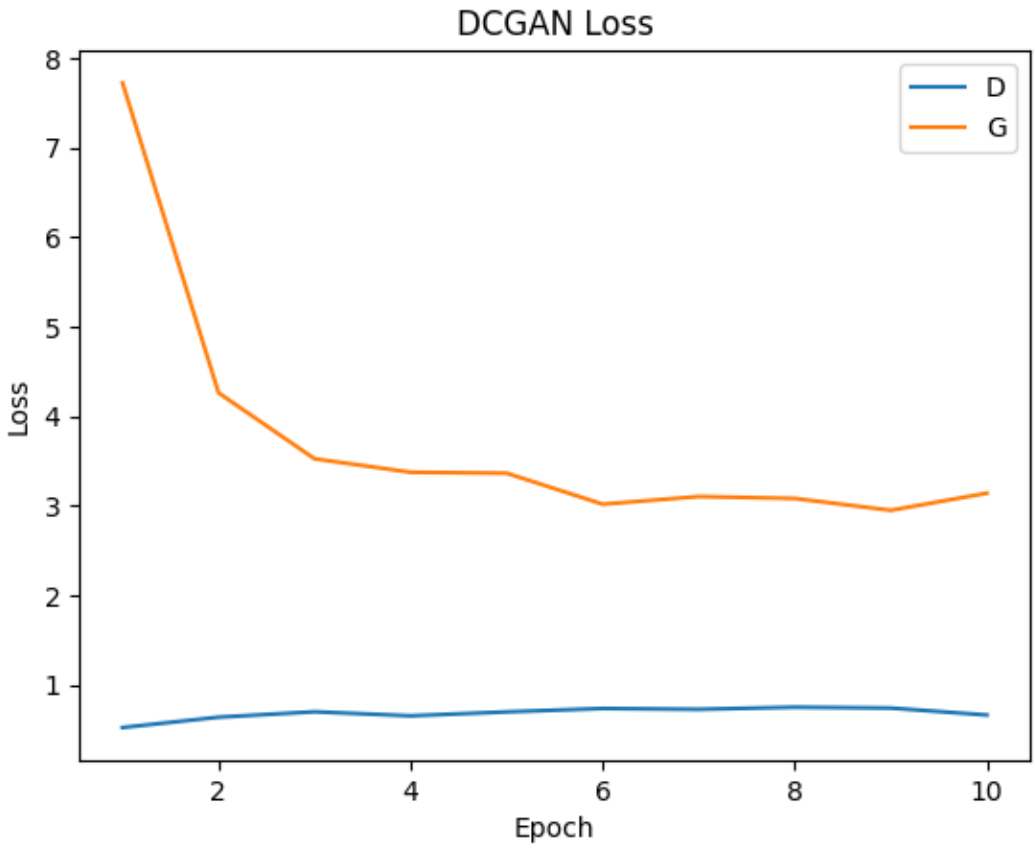
WGAN-GP (if used):

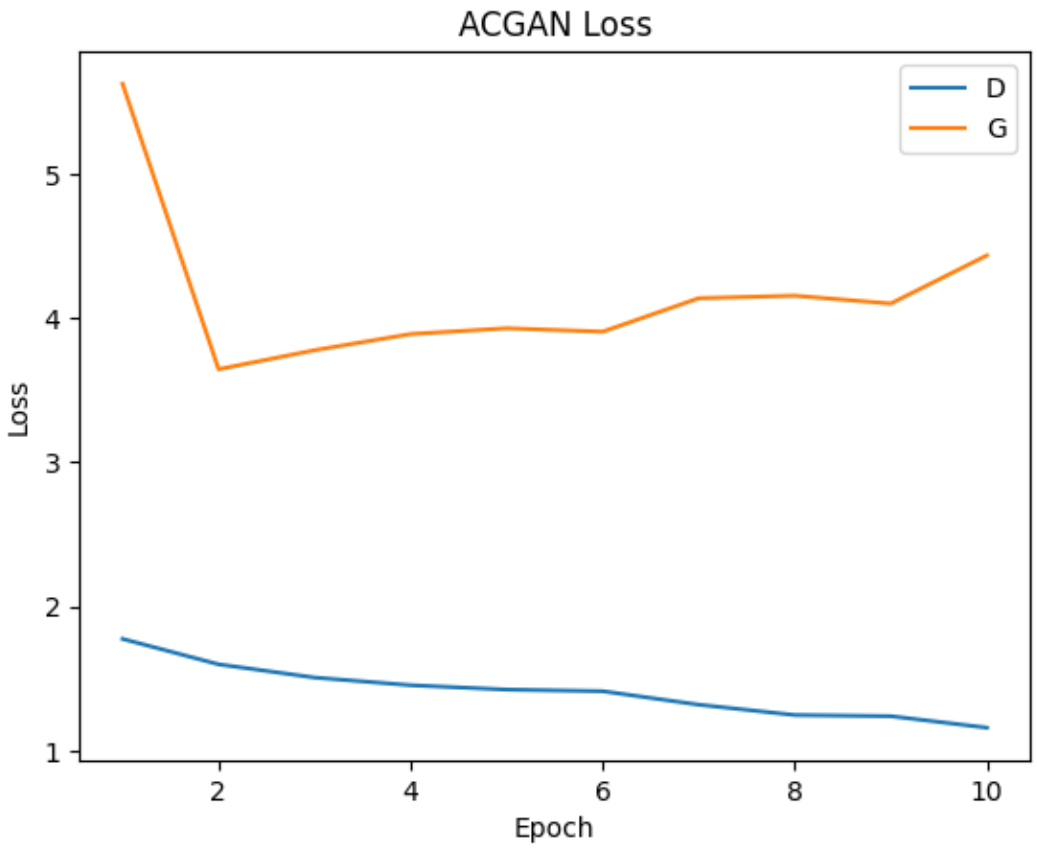
- Most stable training overall.
- Least prone to mode collapse.

ACGAN:

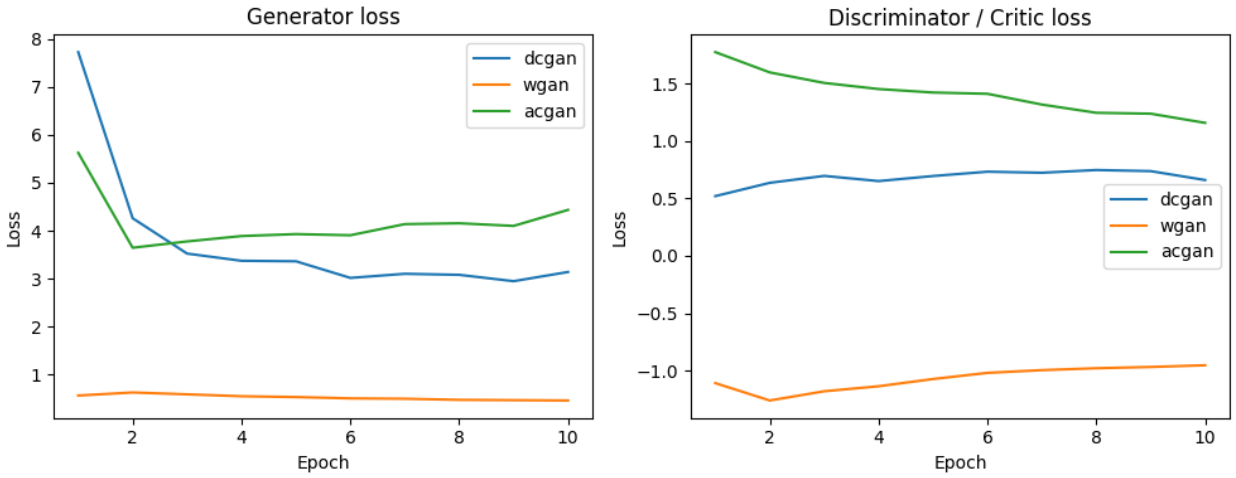
- Conditioned samples follow class structure.
- Quality may vary depending on class; often slightly noisier.

6.2 Loss Behavior





Combined Loss curve –



- DCGAN losses oscillate frequently.
- WGAN critic loss shows smoother convergence.
- ACGAN discriminator splits stability across real/fake + class prediction.

6.3 Stability

- DCGAN: Moderately stable, occasional collapse.
- WGAN: Very stable with clipping.
- WGAN-GP: Most stable.
- ACGAN: Stable if class distribution well-balanced.

---

7. Results Summary Table

Model	Training Stability	Generator Loss Trend	Discriminator / Critic Loss Trend	Image Quality (Epoch ~10)	Notes
DCGAN	Moderate (oscillates as expected)	Drops sharply early, then fluctuates	Oscillatory but stable	Recognizable but slightly blurry images	Good baseline but sensitive to hyperparameters
WGAN	Very high stability	Smooth, monotonic decrease	Critic loss is stable and consistent	Cleaner textures and less noise	Best stability due to Wasserstein loss
ACGAN	High stability (due to class supervision)	Smooth, consistent decline	Higher values since D learns class + real/fake	Most structured and class-consistent images	Best visual quality overall

8. Discussion

Summarize key observations, for example:

- WGAN (or WGAN-GP) generally outperforms DCGAN in stability and realism.
- ACGAN adds class control at the cost of slightly more complex training.
- Model performance can vary heavily by seeds, hyperparameters, and training time.

---

9. Conclusion

- Successfully trained DCGAN, WGAN, and ACGAN.
- Generated samples demonstrate improvements from DCGAN → WGAN.
- ACGAN adds interpretability via class control.
