Machine Learning
LABORATORY: GAN Homework

NAME:	STUDENT ID#:

Objectives:

- Reuse and adapt GAN code to new datasets **(FashionMNIST and CIFAR-10).**
- Generate realistic samples using both standard GAN and **CycleGAN** architectures.
- Explore and compare how different GAN variants handle multi-class image generation tasks.
- Visualize and analyze **mode diversity** and **style mimicry** in generated outputs.
- Discuss the different between GAN and CycleGAN when applied to real-world datasets.

Instructions:

In this homework, you will build on the in-class GAN assignment and extend it in three main ways:

★ **Part 1: Train a GAN on New Datasets (FashionMNIST & CIFAR-10)**

- Choose **at least three classes** from **each dataset**.
  - FashionMNIST examples: shirts (1), trousers (0), coats (4), etc.
  - CIFAR-10 examples: dogs (5), cats (3), airplanes (0), etc.
- Use your in-class GAN architecture to:
  - Train the GAN on each individual class (class-conditional via filtering).
  - Visualize **real**, **fake**, and **mimic** images for each class.
- Train **one model per class**, or create a unified batch-based loader for multi-class training.

★ **Part 2: Implement CycleGAN for Same Tasks**

- Adapt a basic **CycleGAN-style** generator and discriminator.
- Train CycleGAN on the **same three classes** per dataset.
- For each class:
  - Visualize real, fake, and mimic images.
  - Optionally: try one-directional mapping (e.g., fake → real) or cycle consistency.

★ **Part 3: Compare GAN vs CycleGAN**

- Pick a visual or quantitative metric (e.g., visual diversity, sharpness, mimic accuracy).
- Discuss differences in output quality:
  - Which model preserves the structure of the class better?
  - Which one is more stable to train?
  - How do the mimic results compare?
- Include at least **3 visual side-by-side comparisons** in your submission.

For CycleGAN implementation, you may refer to this link: https://junyanz.github.io/CycleGAN/

Lecture: Prof. Hsien-I Lin
TA: Satrio Sanjaya and Muhammad Ahsan

Code Template.	
Step	Procedure
1	(code below)

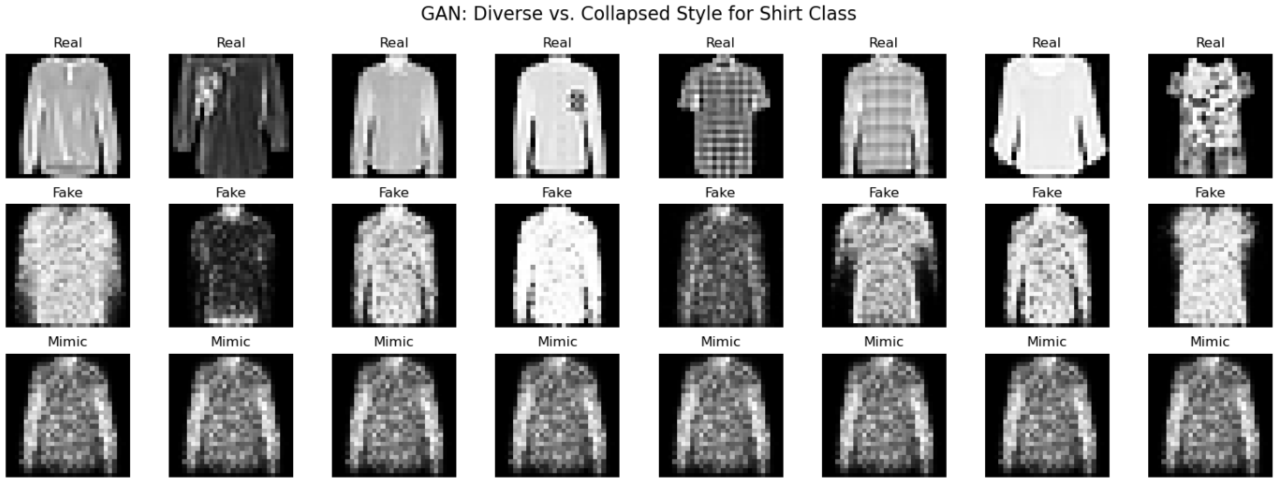
```
#Load Dataset
import torch
import torch.nn as nn
import torch.optim as optim
from torchvision import datasets, transforms
from torch.utils.data import DataLoader, Subset
from torchvision.utils import save_image
import matplotlib.pyplot as plt
import numpy as np

# Setup
z_dim = 100
batch_size = 64
num_epochs = 100
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
img_channels, img_size = 1, 28

# Load only 'shirt' class (FashionMNIST label 1, adjust the label to use another class)
transform = transforms.Compose([
    transforms.ToTensor(),
    transforms.Normalize((0.5,), (0.5,))
])
dataset = datasets.FashionMNIST("./data", train=True, download=True,
transform=transform)
shirt_indices = [i for i, (_, label) in enumerate(dataset) if label == 1]
shirt_dataset = Subset(dataset, shirt_indices[:5000])
loader = DataLoader(shirt_dataset, batch_size=batch_size, shuffle=True)
```

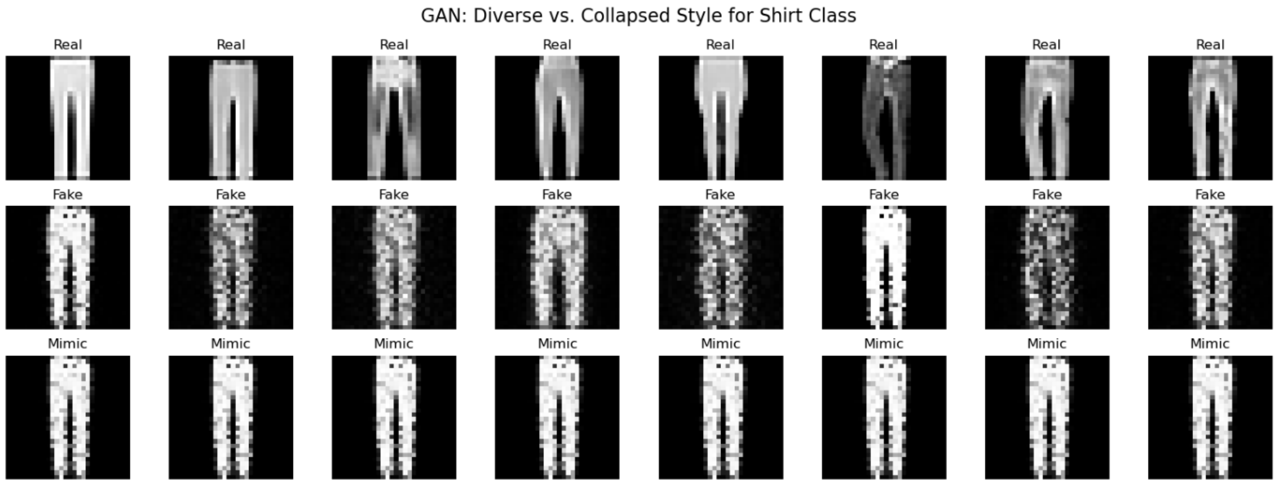
2	(code below)

```
# ====== Generator and Discriminator Definitions ======
# Define the Generator
class Generator(nn.Module):
    def __init__(self, z_dim=100, img_dim=784):
        super().__init__()
        self.gen = nn.Sequential(
            # Define your generator architecture here
        )

    def forward(self, x):
        return self.gen(x)


# Define the Discriminator
class Discriminator(nn.Module):
    def __init__(self, img_dim=784):
        super().__init__()
        self.disc = nn.Sequential(
            # Define your discriminator architecture here
```

	) def forward(self, x): return self.disc(x)
3	# ========== Training Setup ========== # Initialize networks and optimizers, you can adjust the parameters z_dim = 100 lr = 0.0002 gen = Generator(z_dim) disc = Discriminator() criterion = nn.BCELoss() opt_gen = optim.Adam(gen.parameters(), lr) opt_disc = optim.Adam(disc.parameters(), lr)
4	# Write training loop with GAN adversarial loss here # TODO: implement training loop with real/fake labels, forward passes, and optim steps

Example Output:





Grading Assignment & Submission (70% Max)

Implementation (50%):

1. **(15%) GAN on FashionMNIST and CIFAR-10**
   - Trained using at least **3 different classes** per dataset
   - Includes proper visualizations of real, fake, and mimic rows for each class
2. **(15%) CycleGAN on the same classes and datasets**
   - Generator and discriminator implemented with cycle consistency
   - CycleGAN training loop and outputs visualized
3. **(10%) Mimic Mode Implementation**
   - Optimize latent space (GAN) or input image (CycleGAN) to mimic a target real image
   - Generates multiple mimic samples showing collapse to a specific style
4. **(10%) Comparison: GAN vs CycleGAN**
   - At least **3 visual comparisons** between GAN and CycleGAN outputs for the same class
   - May include image sharpness, diversity, or mimic effectiveness

Questions (20%):

1. **(7%) Which model generated more realistic or varied results — GAN or CycleGAN?**
   - Include reasoning with visual proof (e.g., samples or diversity)
2. **(7%) How did style mimic perform across both models?**
   - Did one model preserve class/style better than the other?
   - What were the limitations or artifacts you observed?
3. **(6%) How would you improve the quality of generated results?**
   - Consider architecture, training tricks, normalization, or loss tuning

Submission:

1. Report: Answer all conceptual questions. Include screenshots of your results in the last pages of this PDF File.
2. Code: Submit your complete Python script in either .py or .ipynb format.
3. Upload both your report and code to the E3 system (**Lab9_Homework_Assignment**). Name your files correctly:
   a. Report: StudentID_Lab9_Homework.pdf
   b. Code: StudentID_Lab9_ Homework.py or StudentID_Lab9_Homework.ipynb
4. Deadline: Sunday, 21:00 PM
5. Plagiarism is **strictly prohibited**. Submitting copied work from other students will result in penalties.

Answer:

Using device: cuda

Files already downloaded and verified

Training GAN for FashionMNIST class 0: T-shirt/top

Starting GAN training for Fashion - Class: T-shirt/top

[0/50][0/79] Loss_D: 1.5265 Loss_G: 1.0003

[0/50][50/79] Loss_D: 0.1712 Loss_G: 6.2895

[1/50][0/79] Loss_D: 0.0497 Loss_G: 5.3632

[1/50][50/79] Loss_D: 0.0767 Loss_G: 5.6202

[2/50][0/79] Loss_D: 1.2779 Loss_G: 4.1496

[2/50][50/79] Loss_D: 0.6559 Loss_G: 1.8789

[3/50][0/79] Loss_D: 0.7185 Loss_G: 1.3681

[3/50][50/79] Loss_D: 0.7222 Loss_G: 2.1011

[4/50][0/79] Loss_D: 0.6477 Loss_G: 1.7467

[4/50][50/79] Loss_D: 0.7899 Loss_G: 1.2788

[5/50][0/79] Loss_D: 0.6134 Loss_G: 2.2318

[5/50][50/79] Loss_D: 0.7920 Loss_G: 3.0429

[6/50][0/79] Loss_D: 0.1969 Loss_G: 3.2988

[6/50][50/79] Loss_D: 0.0984 Loss_G: 4.2324

[7/50][0/79] Loss_D: 0.1504 Loss_G: 4.2743

[7/50][50/79] Loss_D: 1.2822 Loss_G: 3.9982

[8/50][0/79] Loss_D: 0.4434 Loss_G: 3.4690

[8/50][50/79] Loss_D: 0.1478 Loss_G: 4.5678

[9/50][0/79] Loss_D: 0.3097 Loss_G: 1.6325

[9/50][50/79] Loss_D: 0.3165 Loss_G: 2.7040

[10/50][0/79] Loss_D: 0.0680 Loss_G: 4.1965

[10/50][50/79] Loss_D: 0.3561 Loss_G: 2.0236

[11/50][0/79] Loss_D: 0.1388 Loss_G: 6.2819

[11/50][50/79] Loss_D: 0.5866 Loss_G: 3.1181

[12/50][0/79] Loss_D: 0.0534 Loss_G: 6.2853

[12/50][50/79] Loss_D: 0.8449 Loss_G: 4.2066

[13/50][0/79] Loss_D: 0.1831 Loss_G: 4.4573

[13/50][50/79] Loss_D: 0.7639 Loss_G: 4.5384

[14/50][0/79] Loss_D: 0.3983 Loss_G: 4.4617

[14/50][50/79] Loss_D: 0.0495 Loss_G: 5.7003

[15/50][0/79] Loss_D: 0.0772 Loss_G: 3.0471

[15/50][50/79] Loss_D: 0.0435 Loss_G: 4.3602

[16/50][0/79] Loss_D: 0.3239 Loss_G: 2.8155

[16/50][50/79] Loss_D: 0.5507 Loss_G: 4.4453

[17/50][0/79] Loss_D: 0.2948 Loss_G: 3.8096

[17/50][50/79] Loss_D: 1.1192 Loss_G: 0.7648

[18/50][0/79] Loss_D: 0.6448 Loss_G: 1.3318

[18/50][50/79] Loss_D: 0.1860 Loss_G: 3.7832

[19/50][0/79] Loss_D: 0.2522 Loss_G: 2.7577

[19/50][50/79] Loss_D: 0.5030 Loss_G: 4.2700

[20/50][0/79] Loss_D: 0.5283 Loss_G: 2.1672

[20/50][50/79] Loss_D: 0.4262 Loss_G: 3.2359

[21/50][0/79] Loss_D: 0.8864 Loss_G: 3.6658

[21/50][50/79] Loss_D: 0.5347 Loss_G: 2.9686

[22/50][0/79] Loss_D: 0.1001 Loss_G: 7.1493

[22/50][50/79] Loss_D: 1.3186 Loss_G: 1.9095

[23/50][0/79] Loss_D: 0.2956 Loss_G: 1.9305

[23/50][50/79] Loss_D: 0.5204 Loss_G: 2.0713

[24/50][0/79] Loss_D: 0.0485 Loss_G: 7.8243

[24/50][50/79] Loss_D: 0.0557 Loss_G: 4.6447

[25/50][0/79] Loss_D: 0.0431 Loss_G: 4.8764

[25/50][50/79] Loss_D: 0.2251 Loss_G: 3.1114

[26/50][0/79] Loss_D: 0.0799 Loss_G: 4.4257

[26/50][50/79] Loss_D: 0.2011 Loss_G: 3.6286

[27/50][0/79] Loss_D: 0.0178 Loss_G: 7.5956

[27/50][50/79] Loss_D: 0.0399 Loss_G: 5.7136

[28/50][0/79] Loss_D: 0.7334 Loss_G: 1.6200

[28/50][50/79] Loss_D: 0.3431 Loss_G: 2.9702

[29/50][0/79] Loss_D: 0.1009 Loss_G: 6.4265

[29/50][50/79] Loss_D: 0.3409 Loss_G: 5.1208

[30/50][0/79] Loss_D: 0.9967 Loss_G: 1.5645

[30/50][50/79] Loss_D: 0.2170 Loss_G: 3.7857

[31/50][0/79] Loss_D: 0.0676 Loss_G: 2.9848

[31/50][50/79] Loss_D: 0.1045 Loss_G: 3.2671

[32/50][0/79] Loss_D: 0.0683 Loss_G: 3.8906

[32/50][50/79] Loss_D: 1.5245 Loss_G: 0.3051

[33/50][0/79] Loss_D: 0.1541 Loss_G: 3.8455

[33/50][50/79] Loss_D: 0.1061 Loss_G: 3.7518

[34/50][0/79] Loss_D: 0.0212 Loss_G: 6.0422

[34/50][50/79] Loss_D: 0.0298 Loss_G: 4.6727

[35/50][0/79] Loss_D: 0.0126 Loss_G: 6.7721

[35/50][50/79] Loss_D: 0.6432 Loss_G: 2.6473

[36/50][0/79] Loss_D: 0.0783 Loss_G: 3.1285

[36/50][50/79] Loss_D: 0.0751 Loss_G: 3.9554

[37/50][0/79] Loss_D: 0.4511 Loss_G: 1.9597

[37/50][50/79] Loss_D: 0.2273 Loss_G: 3.6596

[38/50][0/79] Loss_D: 0.1792 Loss_G: 3.2823

[38/50][50/79] Loss_D: 0.1895 Loss_G: 4.0145

[39/50][0/79] Loss_D: 0.0224 Loss_G: 5.9940

[39/50][50/79] Loss_D: 0.0446 Loss_G: 5.7008

[40/50][0/79] Loss_D: 0.0090 Loss_G: 7.9106

[40/50][50/79] Loss_D: 0.8348 Loss_G: 1.8986

[41/50][0/79] Loss_D: 1.0512 Loss_G: 0.6658

[41/50][50/79] Loss_D: 0.7035 Loss_G: 1.7176

[42/50][0/79] Loss_D: 0.7593 Loss_G: 3.7181

[42/50][50/79] Loss_D: 0.4635 Loss_G: 2.1719

[43/50][0/79] Loss_D: 0.5286 Loss_G: 2.6264

[43/50][50/79] Loss_D: 0.4251 Loss_G: 2.1357

[44/50][0/79] Loss_D: 0.4686 Loss_G: 2.2048

[44/50][50/79] Loss_D: 0.4624 Loss_G: 3.2658

[45/50][0/79] Loss_D: 0.4410 Loss_G: 2.6224

[45/50][50/79] Loss_D: 0.2706 Loss_G: 3.1533

[46/50][0/79] Loss_D: 1.9131 Loss_G: 0.3210

[46/50][50/79] Loss_D: 0.1956 Loss_G: 3.9612

[47/50][0/79] Loss_D: 0.0160 Loss_G: 7.1173

[47/50][50/79] Loss_D: 0.1719 Loss_G: 5.7915

[48/50][0/79] Loss_D: 0.0165 Loss_G: 5.1611

[48/50][50/79] Loss_D: 0.0665 Loss_G: 4.5740

[49/50][0/79] Loss_D: 1.4863 Loss_G: 5.2060

[49/50][50/79] Loss_D: 0.5047 Loss_G: 2.4049

Training GAN for FashionMNIST class 1: Trouser

Starting GAN training for Fashion - Class: Trouser

[0/50][0/79] Loss_D: 1.2774 Loss_G: 1.1273

[0/50][50/79] Loss_D: 0.0253 Loss_G: 4.9918

[1/50][0/79] Loss_D: 0.0111 Loss_G: 5.6362

[1/50][50/79] Loss_D: 0.0067 Loss_G: 6.2845

[2/50][0/79] Loss_D: 0.0069 Loss_G: 6.1470

[2/50][50/79] Loss_D: 0.0035 Loss_G: 6.7432

[3/50][0/79] Loss_D: 0.0014 Loss_G: 6.9629

[3/50][50/79] Loss_D: 0.0010 Loss_G: 7.2604

[4/50][0/79] Loss_D: 0.0020 Loss_G: 7.1276

[4/50][50/79] Loss_D: 0.0011 Loss_G: 7.3658

[5/50][0/79] Loss_D: 0.0011 Loss_G: 7.7627

[5/50][50/79] Loss_D: 0.0008 Loss_G: 7.5768

[6/50][0/79] Loss_D: 0.0006 Loss_G: 7.9195

[6/50][50/79] Loss_D: 0.0004 Loss_G: 8.3001

[7/50][0/79] Loss_D: 0.0006 Loss_G: 8.0720

[7/50][50/79] Loss_D: 0.0003 Loss_G: 8.5994

[8/50][0/79] Loss_D: 0.0003 Loss_G: 8.5497

[8/50][50/79] Loss_D: 0.1993 Loss_G: 9.4981

[9/50][0/79] Loss_D: 1.2821 Loss_G: 9.3776

[9/50][50/79] Loss_D: 1.6087 Loss_G: 4.2425

[10/50][0/79] Loss_D: 1.1465 Loss_G: 1.2252

[10/50][50/79] Loss_D: 0.7713 Loss_G: 1.7734

[11/50][0/79] Loss_D: 1.3513 Loss_G: 1.0118

[11/50][50/79] Loss_D: 1.1396 Loss_G: 0.8416

[12/50][0/79] Loss_D: 1.4201 Loss_G: 1.4493

[12/50][50/79] Loss_D: 1.2446 Loss_G: 1.0492

[13/50][0/79] Loss_D: 1.2296 Loss_G: 1.3361

[13/50][50/79] Loss_D: 1.4478 Loss_G: 1.6375

[14/50][0/79] Loss_D: 2.2375 Loss_G: 1.3274

[14/50][50/79] Loss_D: 1.1686 Loss_G: 1.4393

[15/50][0/79] Loss_D: 0.6755 Loss_G: 1.2208

[15/50][50/79] Loss_D: 1.1763 Loss_G: 1.3805

[16/50][0/79] Loss_D: 0.4523 Loss_G: 1.5619

[16/50][50/79] Loss_D: 1.1800 Loss_G: 1.0948

[17/50][0/79] Loss_D: 0.4019 Loss_G: 3.3614

[17/50][50/79] Loss_D: 1.1600 Loss_G: 0.9819

[18/50][0/79] Loss_D: 1.1298 Loss_G: 2.1698

[18/50][50/79] Loss_D: 0.4864 Loss_G: 2.1346

[19/50][0/79] Loss_D: 0.7061 Loss_G: 4.2822

[19/50][50/79] Loss_D: 0.0829 Loss_G: 4.0437

[20/50][0/79] Loss_D: 0.0592 Loss_G: 4.0685

[20/50][50/79] Loss_D: 0.7970 Loss_G: 1.9291

[21/50][0/79] Loss_D: 0.4180 Loss_G: 1.7006

[21/50][50/79] Loss_D: 0.4464 Loss_G: 2.8687

[22/50][0/79] Loss_D: 0.2099 Loss_G: 2.5146

[22/50][50/79] Loss_D: 1.0621 Loss_G: 3.8940

[23/50][0/79] Loss_D: 0.1835 Loss_G: 2.0080

[23/50][50/79] Loss_D: 0.6722 Loss_G: 1.3980

[24/50][0/79] Loss_D: 0.5646 Loss_G: 2.6877

[24/50][50/79] Loss_D: 0.6719 Loss_G: 1.9696

[25/50][0/79] Loss_D: 0.2575 Loss_G: 2.3421

[25/50][50/79] Loss_D: 0.9031 Loss_G: 1.9790

[26/50][0/79] Loss_D: 0.0682 Loss_G: 4.1228

[26/50][50/79] Loss_D: 0.5948 Loss_G: 2.3591

[27/50][0/79] Loss_D: 1.2691 Loss_G: 1.1300

[27/50][50/79] Loss_D: 0.3516 Loss_G: 3.1678

[28/50][0/79] Loss_D: 1.4176 Loss_G: 3.1216

[28/50][50/79] Loss_D: 0.2194 Loss_G: 3.2084

[29/50][0/79] Loss_D: 2.7921 Loss_G: 0.9051

[29/50][50/79] Loss_D: 0.8603 Loss_G: 1.6874

[30/50][0/79] Loss_D: 0.3072 Loss_G: 2.3548

[30/50][50/79] Loss_D: 0.1572 Loss_G: 3.1081

[31/50][0/79] Loss_D: 0.8537 Loss_G: 3.5988

[31/50][50/79] Loss_D: 0.5014 Loss_G: 2.1313

[32/50][0/79] Loss_D: 0.5259 Loss_G: 3.5357

[32/50][50/79] Loss_D: 0.3805 Loss_G: 1.9644

[33/50][0/79] Loss_D: 0.3712 Loss_G: 3.2942

[33/50][50/79] Loss_D: 0.5019 Loss_G: 2.7275

[34/50][0/79] Loss_D: 0.1052 Loss_G: 3.4245

[34/50][50/79] Loss_D: 0.5473 Loss_G: 2.3844

[35/50][0/79] Loss_D: 0.0492 Loss_G: 3.3470

[35/50][50/79] Loss_D: 0.9409 Loss_G: 1.0486

[36/50][0/79] Loss_D: 0.4028 Loss_G: 2.2602

[36/50][50/79] Loss_D: 0.6337 Loss_G: 3.7143

[37/50][0/79] Loss_D: 1.9966 Loss_G: 5.1783

[37/50][50/79] Loss_D: 0.5398 Loss_G: 2.5038

[38/50][0/79] Loss_D: 0.5441 Loss_G: 3.9786

[38/50][50/79] Loss_D: 0.2560 Loss_G: 3.7490

[39/50][0/79] Loss_D: 0.2618 Loss_G: 2.6960

[39/50][50/79] Loss_D: 0.3139 Loss_G: 3.2726

[40/50][0/79] Loss_D: 0.1176 Loss_G: 3.7725

[40/50][50/79] Loss_D: 0.7726 Loss_G: 1.6915

[41/50][0/79] Loss_D: 0.6065 Loss_G: 3.4299

[41/50][50/79] Loss_D: 0.1101 Loss_G: 3.8790

[42/50][0/79] Loss_D: 0.2634 Loss_G: 3.2733

[42/50][50/79] Loss_D: 0.3395 Loss_G: 4.0133

[43/50][0/79] Loss_D: 0.1609 Loss_G: 3.2951

[43/50][50/79] Loss_D: 0.6142 Loss_G: 2.2229

[44/50][0/79] Loss_D: 0.1793 Loss_G: 3.3059

[44/50][50/79] Loss_D: 1.0841 Loss_G: 5.7546

[45/50][0/79] Loss_D: 0.6724 Loss_G: 2.0315

[45/50][50/79] Loss_D: 0.1082 Loss_G: 3.1962

[46/50][0/79] Loss_D: 0.1915 Loss_G: 2.8176

[46/50][50/79] Loss_D: 0.1022 Loss_G: 3.5984

[47/50][0/79] Loss_D: 0.0925 Loss_G: 7.2516

[47/50][50/79] Loss_D: 1.4667 Loss_G: 4.3404

[48/50][0/79] Loss_D: 1.7098 Loss_G: 0.2623

[48/50][50/79] Loss_D: 0.2865 Loss_G: 2.8312

[49/50][0/79] Loss_D: 0.0407 Loss_G: 4.7798

[49/50][50/79] Loss_D: 0.1102 Loss_G: 3.6396

Training GAN for FashionMNIST class 4: Coat

Starting GAN training for Fashion - Class: Coat

[0/50][0/79] Loss_D: 1.5515 Loss_G: 1.1809

[0/50][50/79] Loss_D: 0.0260 Loss_G: 5.5199

[1/50][0/79] Loss_D: 0.0316 Loss_G: 5.8802

[1/50][50/79] Loss_D: 0.4964 Loss_G: 4.7706

[2/50][0/79] Loss_D: 0.1518 Loss_G: 5.3281

[2/50][50/79] Loss_D: 0.8343 Loss_G: 0.9199

[3/50][0/79] Loss_D: 1.0215 Loss_G: 1.6927

[3/50][50/79] Loss_D: 1.7042 Loss_G: 1.8388

[4/50][0/79] Loss_D: 1.1101 Loss_G: 0.8324

[4/50][50/79] Loss_D: 0.6453 Loss_G: 1.7884

[5/50][0/79] Loss_D: 1.1450 Loss_G: 0.9544

[5/50][50/79] Loss_D: 1.4041 Loss_G: 2.4447

[6/50][0/79] Loss_D: 1.7211 Loss_G: 1.8697

[6/50][50/79] Loss_D: 1.0520 Loss_G: 1.9864

[7/50][0/79] Loss_D: 0.6689 Loss_G: 1.8853

[7/50][50/79] Loss_D: 0.9699 Loss_G: 1.5142

[8/50][0/79] Loss_D: 0.7097 Loss_G: 2.2224

[8/50][50/79] Loss_D: 0.9281 Loss_G: 1.0432

[9/50][0/79] Loss_D: 1.0281 Loss_G: 1.4662

[9/50][50/79] Loss_D: 1.1874 Loss_G: 1.9764

[10/50][0/79] Loss_D: 1.1074 Loss_G: 1.1672

[10/50][50/79] Loss_D: 1.5317 Loss_G: 2.5928

[11/50][0/79] Loss_D: 1.1120 Loss_G: 1.4420

[11/50][50/79] Loss_D: 0.6655 Loss_G: 2.9031

[12/50][0/79] Loss_D: 0.9055 Loss_G: 2.5976

[12/50][50/79] Loss_D: 0.6841 Loss_G: 1.4532

[13/50][0/79] Loss_D: 0.1412 Loss_G: 2.9259

[13/50][50/79] Loss_D: 0.6917 Loss_G: 1.1920

[14/50][0/79] Loss_D: 1.3145 Loss_G: 3.0795

[14/50][50/79] Loss_D: 0.5620 Loss_G: 1.8004

[15/50][0/79] Loss_D: 1.1662 Loss_G: 3.1625

[15/50][50/79] Loss_D: 0.4754 Loss_G: 3.0385

[16/50][0/79] Loss_D: 0.2891 Loss_G: 2.4172

[16/50][50/79] Loss_D: 0.9886 Loss_G: 1.2505

[17/50][0/79] Loss_D: 0.7876 Loss_G: 2.4043

[17/50][50/79] Loss_D: 0.7435 Loss_G: 2.4591

[18/50][0/79] Loss_D: 1.0541 Loss_G: 3.4093

[18/50][50/79] Loss_D: 0.4075 Loss_G: 1.7214

[19/50][0/79] Loss_D: 0.4486 Loss_G: 2.0885

[19/50][50/79] Loss_D: 1.2460 Loss_G: 6.5148

[20/50][0/79] Loss_D: 0.4427 Loss_G: 2.5224

[20/50][50/79] Loss_D: 0.6051 Loss_G: 4.6899

[21/50][0/79] Loss_D: 0.5110 Loss_G: 2.1604

[21/50][50/79] Loss_D: 0.5169 Loss_G: 3.1037

[22/50][0/79] Loss_D: 0.1795 Loss_G: 2.5352

[22/50][50/79] Loss_D: 0.2620 Loss_G: 3.0108

[23/50][0/79] Loss_D: 0.1995 Loss_G: 3.2696

[23/50][50/79] Loss_D: 0.1861 Loss_G: 3.2786

[24/50][0/79] Loss_D: 1.4969 Loss_G: 3.9230

[24/50][50/79] Loss_D: 1.5909 Loss_G: 2.3166

[25/50][0/79] Loss_D: 1.0019 Loss_G: 1.0063

[25/50][50/79] Loss_D: 0.2234 Loss_G: 2.8129

[26/50][0/79] Loss_D: 0.1821 Loss_G: 2.9284

[26/50][50/79] Loss_D: 0.0955 Loss_G: 3.5876

[27/50][0/79] Loss_D: 0.6664 Loss_G: 2.7864

[27/50][50/79] Loss_D: 0.0958 Loss_G: 3.7333

[28/50][0/79] Loss_D: 0.1009 Loss_G: 3.5968

[28/50][50/79] Loss_D: 1.4897 Loss_G: 1.9720

[29/50][0/79] Loss_D: 0.7639 Loss_G: 1.8972

[29/50][50/79] Loss_D: 0.2314 Loss_G: 2.8364

[30/50][0/79] Loss_D: 0.2468 Loss_G: 2.6130

[30/50][50/79] Loss_D: 0.4104 Loss_G: 4.0778

[31/50][0/79] Loss_D: 0.1313 Loss_G: 3.6776

[31/50][50/79] Loss_D: 0.1237 Loss_G: 3.2000

[32/50][0/79] Loss_D: 0.0836 Loss_G: 3.6086

[32/50][50/79] Loss_D: 0.0328 Loss_G: 4.3815

[33/50][0/79] Loss_D: 0.7912 Loss_G: 1.5538

[33/50][50/79] Loss_D: 0.1573 Loss_G: 3.4286

[34/50][0/79] Loss_D: 0.1498 Loss_G: 5.9665

[34/50][50/79] Loss_D: 0.0597 Loss_G: 3.9499

[35/50][0/79] Loss_D: 0.9497 Loss_G: 3.4676

[35/50][50/79] Loss_D: 0.2317 Loss_G: 3.0060

[36/50][0/79] Loss_D: 0.1035 Loss_G: 4.1637

[36/50][50/79] Loss_D: 0.8341 Loss_G: 2.5410

[37/50][0/79] Loss_D: 1.5736 Loss_G: 4.7775

[37/50][50/79] Loss_D: 0.6349 Loss_G: 2.6783

[38/50][0/79] Loss_D: 0.1986 Loss_G: 2.9315

[38/50][50/79] Loss_D: 0.1074 Loss_G: 3.3248

[39/50][0/79] Loss_D: 0.7993 Loss_G: 3.7190

[39/50][50/79] Loss_D: 0.0616 Loss_G: 5.7368

[40/50][0/79] Loss_D: 0.1156 Loss_G: 5.8382

[40/50][50/79] Loss_D: 0.0375 Loss_G: 6.1267

[41/50][0/79] Loss_D: 0.0150 Loss_G: 7.3104

[41/50][50/79] Loss_D: 0.0472 Loss_G: 4.8922

[42/50][0/79] Loss_D: 0.0443 Loss_G: 4.7845

[42/50][50/79] Loss_D: 0.6262 Loss_G: 2.5229

[43/50][0/79] Loss_D: 0.7800 Loss_G: 5.9165

[43/50][50/79] Loss_D: 0.3418 Loss_G: 2.7161

[44/50][0/79] Loss_D: 0.0205 Loss_G: 5.1456

[44/50][50/79] Loss_D: 0.0932 Loss_G: 3.2361

[45/50][0/79] Loss_D: 0.0209 Loss_G: 5.5035

[45/50][50/79] Loss_D: 0.0474 Loss_G: 4.0705

[46/50][0/79] Loss_D: 0.0158 Loss_G: 5.4898

[46/50][50/79] Loss_D: 0.0752 Loss_G: 4.2992

[47/50][0/79] Loss_D: 2.2035 Loss_G: 2.1254

[47/50][50/79] Loss_D: 0.7855 Loss_G: 1.8538

[48/50][0/79] Loss_D: 0.9699 Loss_G: 3.3201

[48/50][50/79] Loss_D: 0.6548 Loss_G: 2.0260

[49/50][0/79] Loss_D: 0.8113 Loss_G: 2.1919

[49/50][50/79] Loss_D: 0.9441 Loss_G: 2.6776

Training GAN for CIFAR-10 class 0: Airplane

Starting GAN training for CIFAR - Class: Airplane

[0/50][0/79] Loss_D: 1.5602 Loss_G: 1.3600

[0/50][50/79] Loss_D: 0.1634 Loss_G: 5.3249

[1/50][0/79] Loss_D: 0.1423 Loss_G: 5.9280

[1/50][50/79] Loss_D: 0.5256 Loss_G: 3.4728

[2/50][0/79] Loss_D: 0.3182 Loss_G: 3.0194

[2/50][50/79] Loss_D: 0.4523 Loss_G: 1.8724

[3/50][0/79] Loss_D: 0.4396 Loss_G: 2.0354

[3/50][50/79] Loss_D: 1.1050 Loss_G: 2.2411

[4/50][0/79] Loss_D: 1.6951 Loss_G: 6.2090

[4/50][50/79] Loss_D: 1.0164 Loss_G: 4.0096

[5/50][0/79] Loss_D: 0.3990 Loss_G: 3.5309

[5/50][50/79] Loss_D: 0.4626 Loss_G: 3.6265

[6/50][0/79] Loss_D: 1.0184 Loss_G: 1.1392

[6/50][50/79] Loss_D: 0.6201 Loss_G: 1.8005

[7/50][0/79] Loss_D: 0.4513 Loss_G: 1.9163

[7/50][50/79] Loss_D: 0.6508 Loss_G: 2.7764

[8/50][0/79] Loss_D: 0.4509 Loss_G: 2.8311

[8/50][50/79] Loss_D: 0.4483 Loss_G: 3.2507

[9/50][0/79] Loss_D: 0.6640 Loss_G: 3.0862

[9/50][50/79] Loss_D: 0.3088 Loss_G: 3.3151

[10/50][0/79] Loss_D: 1.0436 Loss_G: 5.3147

[10/50][50/79] Loss_D: 0.5071 Loss_G: 2.3179

[11/50][0/79] Loss_D: 1.7164 Loss_G: 8.0190

[11/50][50/79] Loss_D: 0.2732 Loss_G: 2.9117

[12/50][0/79] Loss_D: 0.1753 Loss_G: 3.0633

[12/50][50/79] Loss_D: 0.2658 Loss_G: 2.6855

[13/50][0/79] Loss_D: 2.1407 Loss_G: 7.6513

[13/50][50/79] Loss_D: 0.7701 Loss_G: 4.4644

[14/50][0/79] Loss_D: 0.2635 Loss_G: 3.3920

[14/50][50/79] Loss_D: 0.5736 Loss_G: 3.2310

[15/50][0/79] Loss_D: 1.3103 Loss_G: 5.4543

[15/50][50/79] Loss_D: 0.6609 Loss_G: 1.7316

[16/50][0/79] Loss_D: 0.5779 Loss_G: 2.5942

[16/50][50/79] Loss_D: 0.3917 Loss_G: 2.2303

[17/50][0/79] Loss_D: 0.8446 Loss_G: 2.6494

[17/50][50/79] Loss_D: 0.4072 Loss_G: 2.7680

[18/50][0/79] Loss_D: 0.2276 Loss_G: 3.7908

[18/50][50/79] Loss_D: 0.4487 Loss_G: 3.5834

[19/50][0/79] Loss_D: 0.5586 Loss_G: 2.5543

[19/50][50/79] Loss_D: 0.2546 Loss_G: 2.6404

[20/50][0/79] Loss_D: 1.4865 Loss_G: 5.7742

[20/50][50/79] Loss_D: 0.3265 Loss_G: 2.9476

[21/50][0/79] Loss_D: 1.1971 Loss_G: 2.2073

[21/50][50/79] Loss_D: 0.2546 Loss_G: 3.2938

[22/50][0/79] Loss_D: 0.4726 Loss_G: 3.1417

[22/50][50/79] Loss_D: 0.5937 Loss_G: 2.2244

[23/50][0/79] Loss_D: 1.7154 Loss_G: 5.3251

[23/50][50/79] Loss_D: 0.3469 Loss_G: 2.9447

[24/50][0/79] Loss_D: 0.7393 Loss_G: 3.0625

[24/50][50/79] Loss_D: 0.5423 Loss_G: 2.4225

[25/50][0/79] Loss_D: 0.8578 Loss_G: 2.4683

[25/50][50/79] Loss_D: 0.3121 Loss_G: 3.3018

[26/50][0/79] Loss_D: 0.5381 Loss_G: 2.6589

[26/50][50/79] Loss_D: 0.2825 Loss_G: 2.7747

[27/50][0/79] Loss_D: 0.4944 Loss_G: 2.7212

[27/50][50/79] Loss_D: 0.6665 Loss_G: 3.7755

[28/50][0/79] Loss_D: 0.5336 Loss_G: 1.8563

[28/50][50/79] Loss_D: 0.2223 Loss_G: 3.2227

[29/50][0/79] Loss_D: 0.1739 Loss_G: 2.8575

[29/50][50/79] Loss_D: 0.7885 Loss_G: 3.8139

[30/50][0/79] Loss_D: 0.8885 Loss_G: 4.2664

[30/50][50/79] Loss_D: 0.2607 Loss_G: 2.8832

[31/50][0/79] Loss_D: 0.3356 Loss_G: 3.0050

[31/50][50/79] Loss_D: 0.4636 Loss_G: 2.4518

[32/50][0/79] Loss_D: 0.6382 Loss_G: 2.4206

[32/50][50/79] Loss_D: 0.2522 Loss_G: 3.1959

[33/50][0/79] Loss_D: 0.2373 Loss_G: 3.7172

[33/50][50/79] Loss_D: 0.5587 Loss_G: 2.4155

[34/50][0/79] Loss_D: 1.1477 Loss_G: 4.1272

[34/50][50/79] Loss_D: 0.6870 Loss_G: 2.2040

[35/50][0/79] Loss_D: 2.6476 Loss_G: 7.2000

[35/50][50/79] Loss_D: 0.4772 Loss_G: 1.1940

[36/50][0/79] Loss_D: 0.2788 Loss_G: 2.2541

[36/50][50/79] Loss_D: 0.2659 Loss_G: 2.8054

[37/50][0/79] Loss_D: 1.4237 Loss_G: 6.0820

[37/50][50/79] Loss_D: 0.2435 Loss_G: 2.9618

[38/50][0/79] Loss_D: 0.4252 Loss_G: 4.3920

[38/50][50/79] Loss_D: 0.6218 Loss_G: 2.4108

[39/50][0/79] Loss_D: 0.2529 Loss_G: 3.2624

[39/50][50/79] Loss_D: 1.3704 Loss_G: 2.1172

[40/50][0/79] Loss_D: 2.1482 Loss_G: 6.2576

[40/50][50/79] Loss_D: 0.7936 Loss_G: 0.6842

[41/50][0/79] Loss_D: 0.7597 Loss_G: 4.6029

[41/50][50/79] Loss_D: 0.3299 Loss_G: 2.5994

[42/50][0/79] Loss_D: 0.2559 Loss_G: 3.5582

[42/50][50/79] Loss_D: 0.8902 Loss_G: 1.5340

[43/50][0/79] Loss_D: 0.7187 Loss_G: 1.4016

[43/50][50/79] Loss_D: 0.4849 Loss_G: 4.4667

[44/50][0/79] Loss_D: 0.3392 Loss_G: 4.3732

[44/50][50/79] Loss_D: 0.8927 Loss_G: 3.9869

[45/50][0/79] Loss_D: 0.6405 Loss_G: 4.2885

[45/50][50/79] Loss_D: 0.4038 Loss_G: 2.8485

[46/50][0/79] Loss_D: 0.6006 Loss_G: 0.8869

[46/50][50/79] Loss_D: 0.2922 Loss_G: 2.9427

[47/50][0/79] Loss_D: 0.3619 Loss_G: 4.0374

[47/50][50/79] Loss_D: 0.3157 Loss_G: 2.4476

[48/50][0/79] Loss_D: 0.2425 Loss_G: 3.3027

[48/50][50/79] Loss_D: 0.3470 Loss_G: 3.4591

[49/50][0/79] Loss_D: 0.2262 Loss_G: 3.1416

[49/50][50/79] Loss_D: 0.7526 Loss_G: 2.2327

Training GAN for CIFAR-10 class 3: Cat

Starting GAN training for CIFAR - Class: Cat

[0/50][0/79] Loss_D: 1.4672 Loss_G: 1.4547

[0/50][50/79] Loss_D: 0.1317 Loss_G: 5.6020

[1/50][0/79] Loss_D: 0.0620 Loss_G: 6.5839

[1/50][50/79] Loss_D: 0.0851 Loss_G: 7.6636

[2/50][0/79] Loss_D: 0.0878 Loss_G: 5.0774

[2/50][50/79] Loss_D: 0.5334 Loss_G: 4.4213

[3/50][0/79] Loss_D: 0.5961 Loss_G: 3.4433

[3/50][50/79] Loss_D: 0.6758 Loss_G: 4.7081

[4/50][0/79] Loss_D: 1.4267 Loss_G: 4.9390

[4/50][50/79] Loss_D: 0.7970 Loss_G: 2.9185

[5/50][0/79] Loss_D: 1.3232 Loss_G: 2.9474

[5/50][50/79] Loss_D: 0.6011 Loss_G: 2.2060

[6/50][0/79] Loss_D: 0.4932 Loss_G: 2.7815

[6/50][50/79] Loss_D: 1.3486 Loss_G: 2.0531

[7/50][0/79] Loss_D: 0.9296 Loss_G: 5.2288

[7/50][50/79] Loss_D: 0.4138 Loss_G: 4.1905

[8/50][0/79] Loss_D: 1.5398 Loss_G: 5.2976

[8/50][50/79] Loss_D: 0.5610 Loss_G: 2.6310

[9/50][0/79] Loss_D: 0.3660 Loss_G: 2.6763

[9/50][50/79] Loss_D: 0.8810 Loss_G: 2.6768

[10/50][0/79] Loss_D: 0.5088 Loss_G: 2.9992

[10/50][50/79] Loss_D: 0.5173 Loss_G: 2.2382

[11/50][0/79] Loss_D: 1.9672 Loss_G: 1.8563

[11/50][50/79] Loss_D: 0.5715 Loss_G: 2.6877

[12/50][0/79] Loss_D: 0.4934 Loss_G: 2.3633

[12/50][50/79] Loss_D: 1.0648 Loss_G: 2.1628

[13/50][0/79] Loss_D: 0.7202 Loss_G: 2.8112

[13/50][50/79] Loss_D: 0.9698 Loss_G: 2.9376

[14/50][0/79] Loss_D: 0.4954 Loss_G: 2.9865

[14/50][50/79] Loss_D: 0.6902 Loss_G: 2.3152

[15/50][0/79] Loss_D: 0.2990 Loss_G: 2.6920

[15/50][50/79] Loss_D: 0.9399 Loss_G: 2.8742

[16/50][0/79] Loss_D: 1.1032 Loss_G: 4.1858

[16/50][50/79] Loss_D: 0.3179 Loss_G: 2.8145

[17/50][0/79] Loss_D: 0.0923 Loss_G: 3.8544

[17/50][50/79] Loss_D: 0.2812 Loss_G: 3.1561

[18/50][0/79] Loss_D: 0.9984 Loss_G: 1.9669

[18/50][50/79] Loss_D: 0.4012 Loss_G: 2.2152

[19/50][0/79] Loss_D: 0.6443 Loss_G: 2.3128

[19/50][50/79] Loss_D: 0.3292 Loss_G: 2.9538

[20/50][0/79] Loss_D: 0.2495 Loss_G: 3.7724

[20/50][50/79] Loss_D: 0.9332 Loss_G: 2.5346

[21/50][0/79] Loss_D: 0.4210 Loss_G: 3.1647

[21/50][50/79] Loss_D: 0.2491 Loss_G: 3.7280

[22/50][0/79] Loss_D: 0.6878 Loss_G: 2.2018

[22/50][50/79] Loss_D: 0.6018 Loss_G: 2.9670

[23/50][0/79] Loss_D: 0.1403 Loss_G: 4.5027

[23/50][50/79] Loss_D: 0.5248 Loss_G: 4.1154

[24/50][0/79] Loss_D: 0.2495 Loss_G: 2.9045

[24/50][50/79] Loss_D: 0.2321 Loss_G: 2.9734

[25/50][0/79] Loss_D: 2.1748 Loss_G: 5.0985

[25/50][50/79] Loss_D: 0.8305 Loss_G: 2.1585

[26/50][0/79] Loss_D: 0.6808 Loss_G: 2.9007

[26/50][50/79] Loss_D: 0.3081 Loss_G: 2.6298

[27/50][0/79] Loss_D: 0.0919 Loss_G: 3.5876

[27/50][50/79] Loss_D: 0.8209 Loss_G: 3.5011

[28/50][0/79] Loss_D: 0.9700 Loss_G: 3.5186

[28/50][50/79] Loss_D: 0.4505 Loss_G: 3.2449

[29/50][0/79] Loss_D: 0.2482 Loss_G: 3.0999

[29/50][50/79] Loss_D: 0.8350 Loss_G: 3.2946

[30/50][0/79] Loss_D: 1.1301 Loss_G: 6.1032

[30/50][50/79] Loss_D: 0.5875 Loss_G: 3.5220

[31/50][0/79] Loss_D: 0.0590 Loss_G: 4.7810

[31/50][50/79] Loss_D: 0.1407 Loss_G: 6.0627

[32/50][0/79] Loss_D: 0.3353 Loss_G: 2.6845

[32/50][50/79] Loss_D: 0.0873 Loss_G: 3.7386

[33/50][0/79] Loss_D: 0.4576 Loss_G: 3.4472

[33/50][50/79] Loss_D: 0.5056 Loss_G: 3.6642

[34/50][0/79] Loss_D: 0.9507 Loss_G: 3.9872

[34/50][50/79] Loss_D: 0.0783 Loss_G: 4.6267

[35/50][0/79] Loss_D: 0.6530 Loss_G: 2.7402

[35/50][50/79] Loss_D: 1.8625 Loss_G: 3.9857

[36/50][0/79] Loss_D: 0.0891 Loss_G: 4.7614

[36/50][50/79] Loss_D: 0.4266 Loss_G: 4.1311

[37/50][0/79] Loss_D: 0.2711 Loss_G: 4.3003

[37/50][50/79] Loss_D: 0.3276 Loss_G: 3.3996

[38/50][0/79] Loss_D: 0.1766 Loss_G: 3.7906

[38/50][50/79] Loss_D: 0.4264 Loss_G: 3.0109

[39/50][0/79] Loss_D: 0.3147 Loss_G: 2.9342

[39/50][50/79] Loss_D: 0.7867 Loss_G: 4.5117

[40/50][0/79] Loss_D: 0.1097 Loss_G: 3.1245

[40/50][50/79] Loss_D: 0.7451 Loss_G: 2.2846

[41/50][0/79] Loss_D: 0.3102 Loss_G: 2.2978

[41/50][50/79] Loss_D: 0.2322 Loss_G: 2.9913

[42/50][0/79] Loss_D: 0.4438 Loss_G: 2.9461

[42/50][50/79] Loss_D: 0.5431 Loss_G: 1.3182

[43/50][0/79] Loss_D: 0.5589 Loss_G: 3.4700

[43/50][50/79] Loss_D: 0.3642 Loss_G: 2.2013

[44/50][0/79] Loss_D: 0.2418 Loss_G: 3.3721

[44/50][50/79] Loss_D: 0.1304 Loss_G: 3.5497

[45/50][0/79] Loss_D: 0.0331 Loss_G: 4.2679

[45/50][50/79] Loss_D: 0.2659 Loss_G: 4.9303

[46/50][0/79] Loss_D: 0.1644 Loss_G: 3.3932

[46/50][50/79] Loss_D: 0.2640 Loss_G: 3.3269

[47/50][0/79] Loss_D: 0.3422 Loss_G: 4.5049

[47/50][50/79] Loss_D: 0.1145 Loss_G: 4.4373

[48/50][0/79] Loss_D: 1.3757 Loss_G: 5.3267

[48/50][50/79] Loss_D: 0.3355 Loss_G: 2.1589

[49/50][0/79] Loss_D: 0.6676 Loss_G: 4.4245

[49/50][50/79] Loss_D: 0.0715 Loss_G: 5.5405

Training GAN for CIFAR-10 class 5: Dog

Starting GAN training for CIFAR - Class: Dog

[0/50][0/79] Loss_D: 1.2617 Loss_G: 1.5450

[0/50][50/79] Loss_D: 0.0701 Loss_G: 5.2968

[1/50][0/79] Loss_D: 0.0955 Loss_G: 6.0558

[1/50][50/79] Loss_D: 0.0603 Loss_G: 6.5396

[2/50][0/79] Loss_D: 0.0489 Loss_G: 6.1110

[2/50][50/79] Loss_D: 0.6013 Loss_G: 3.0587

[3/50][0/79] Loss_D: 0.6410 Loss_G: 3.0900

[3/50][50/79] Loss_D: 0.6070 Loss_G: 2.6639

[4/50][0/79] Loss_D: 0.5339 Loss_G: 2.5725

[4/50][50/79] Loss_D: 0.7423 Loss_G: 2.5117

[5/50][0/79] Loss_D: 1.0993 Loss_G: 2.7477

[5/50][50/79] Loss_D: 1.0540 Loss_G: 1.8503

[6/50][0/79] Loss_D: 0.8782 Loss_G: 1.9905

[6/50][50/79] Loss_D: 0.6600 Loss_G: 3.0640

[7/50][0/79] Loss_D: 0.6933 Loss_G: 3.0452

[7/50][50/79] Loss_D: 0.6634 Loss_G: 1.5907

[8/50][0/79] Loss_D: 0.5852 Loss_G: 3.1201

[8/50][50/79] Loss_D: 0.5012 Loss_G: 3.2012

[9/50][0/79] Loss_D: 0.7639 Loss_G: 1.9013

[9/50][50/79] Loss_D: 0.9386 Loss_G: 2.9354

[10/50][0/79] Loss_D: 1.0325 Loss_G: 2.8301

[10/50][50/79] Loss_D: 0.6373 Loss_G: 2.1171

[11/50][0/79] Loss_D: 1.4101 Loss_G: 3.8668

[11/50][50/79] Loss_D: 1.1571 Loss_G: 1.3973

[12/50][0/79] Loss_D: 0.8817 Loss_G: 3.0038

[12/50][50/79] Loss_D: 0.5439 Loss_G: 2.4561

[13/50][0/79] Loss_D: 1.4543 Loss_G: 4.6617

[13/50][50/79] Loss_D: 0.4401 Loss_G: 2.3854

[14/50][0/79] Loss_D: 0.6597 Loss_G: 2.1759

[14/50][50/79] Loss_D: 0.4165 Loss_G: 2.9411

[15/50][0/79] Loss_D: 1.1177 Loss_G: 4.2916

[15/50][50/79] Loss_D: 0.5331 Loss_G: 3.0529

[16/50][0/79] Loss_D: 1.0476 Loss_G: 2.9520

[16/50][50/79] Loss_D: 0.4863 Loss_G: 2.1734

[17/50][0/79] Loss_D: 0.2831 Loss_G: 2.9938

[17/50][50/79] Loss_D: 0.5661 Loss_G: 2.2233

[18/50][0/79] Loss_D: 0.8036 Loss_G: 2.6069

[18/50][50/79] Loss_D: 0.4003 Loss_G: 1.9651

[19/50][0/79] Loss_D: 0.2409 Loss_G: 2.5719

[19/50][50/79] Loss_D: 0.2675 Loss_G: 3.0038

[20/50][0/79] Loss_D: 1.3006 Loss_G: 3.9263

[20/50][50/79] Loss_D: 1.0032 Loss_G: 2.4393

[21/50][0/79] Loss_D: 1.2157 Loss_G: 2.8236

[21/50][50/79] Loss_D: 0.6499 Loss_G: 3.1595

[22/50][0/79] Loss_D: 0.4826 Loss_G: 3.7579

[22/50][50/79] Loss_D: 0.1786 Loss_G: 3.3390

[23/50][0/79] Loss_D: 0.3356 Loss_G: 4.7024

[23/50][50/79] Loss_D: 0.1915 Loss_G: 3.0225

[24/50][0/79] Loss_D: 0.2422 Loss_G: 2.7033

[24/50][50/79] Loss_D: 0.1580 Loss_G: 2.4957

[25/50][0/79] Loss_D: 0.3310 Loss_G: 2.1382

[25/50][50/79] Loss_D: 0.1658 Loss_G: 3.2492

[26/50][0/79] Loss_D: 1.0717 Loss_G: 2.8925

[26/50][50/79] Loss_D: 0.3916 Loss_G: 3.1476

[27/50][0/79] Loss_D: 0.9628 Loss_G: 4.5143

[27/50][50/79] Loss_D: 0.2089 Loss_G: 3.7454

[28/50][0/79] Loss_D: 0.3172 Loss_G: 3.7000

[28/50][50/79] Loss_D: 0.3114 Loss_G: 3.1204

[29/50][0/79] Loss_D: 0.2967 Loss_G: 3.5023

[29/50][50/79] Loss_D: 0.8487 Loss_G: 4.8360

[30/50][0/79] Loss_D: 0.2559 Loss_G: 3.3088

[30/50][50/79] Loss_D: 1.5696 Loss_G: 6.4470

[31/50][0/79] Loss_D: 0.2285 Loss_G: 3.3948

[31/50][50/79] Loss_D: 0.1227 Loss_G: 3.2843

[32/50][0/79] Loss_D: 0.1604 Loss_G: 3.9113

[32/50][50/79] Loss_D: 0.4302 Loss_G: 2.9984

[33/50][0/79] Loss_D: 0.0963 Loss_G: 3.2037

[33/50][50/79] Loss_D: 0.5357 Loss_G: 5.3342

[34/50][0/79] Loss_D: 0.0706 Loss_G: 4.1069

[34/50][50/79] Loss_D: 0.3352 Loss_G: 3.8334

[35/50][0/79] Loss_D: 0.0896 Loss_G: 4.0391

[35/50][50/79] Loss_D: 0.2289 Loss_G: 3.4856

[36/50][0/79] Loss_D: 2.3486 Loss_G: 8.6818

[36/50][50/79] Loss_D: 0.3321 Loss_G: 3.8812

[37/50][0/79] Loss_D: 0.1025 Loss_G: 4.2158

[37/50][50/79] Loss_D: 0.2484 Loss_G: 3.1130

[38/50][0/79] Loss_D: 0.2890 Loss_G: 3.2150

[38/50][50/79] Loss_D: 0.1813 Loss_G: 4.0483

[39/50][0/79] Loss_D: 2.1144 Loss_G: 7.2760

[39/50][50/79] Loss_D: 0.0830 Loss_G: 4.5615

[40/50][0/79] Loss_D: 0.0691 Loss_G: 5.7422

[40/50][50/79] Loss_D: 0.1076 Loss_G: 3.6401

[41/50][0/79] Loss_D: 2.0430 Loss_G: 6.7716

[41/50][50/79] Loss_D: 0.2963 Loss_G: 3.7060

[42/50][0/79] Loss_D: 0.1021 Loss_G: 4.1404

[42/50][50/79] Loss_D: 0.1492 Loss_G: 3.2255

[43/50][0/79] Loss_D: 0.6448 Loss_G: 6.0497

[43/50][50/79] Loss_D: 0.1724 Loss_G: 4.0229

[44/50][0/79] Loss_D: 0.2521 Loss_G: 3.0106

[44/50][50/79] Loss_D: 0.2888 Loss_G: 2.9950

[45/50][0/79] Loss_D: 0.1182 Loss_G: 3.4458

[45/50][50/79] Loss_D: 0.1140 Loss_G: 4.3607

[46/50][0/79] Loss_D: 0.3490 Loss_G: 3.1232

[46/50][50/79] Loss_D: 0.1221 Loss_G: 3.9628

[47/50][0/79] Loss_D: 0.0597 Loss_G: 5.4866

[47/50][50/79] Loss_D: 0.2916 Loss_G: 2.4012

[48/50][0/79] Loss_D: 0.1342 Loss_G: 3.9498

[48/50][50/79] Loss_D: 1.0324 Loss_G: 2.8533

[49/50][0/79] Loss_D: 1.3730 Loss_G: 5.2998

[49/50][50/79] Loss_D: 0.3834 Loss_G: 3.1159

Training CycleGAN for FashionMNIST: Trousers to T-shirts

Starting CycleGAN training for Fashion - Classes: Trouser and Tshirt

[0/50][0/79] Loss_D: 1.5213 Loss_G: 24.3425 Loss_cycle: 14.7142

[0/50][50/79] Loss_D: 0.4203 Loss_G: 5.7416 Loss_cycle: 3.4684

[1/50][0/79] Loss_D: 0.4402 Loss_G: 6.6906 Loss_cycle: 3.8243

[1/50][50/79] Loss_D: 0.5320 Loss_G: 5.2279 Loss_cycle: 2.8994

[2/50][0/79] Loss_D: 0.5415 Loss_G: 4.0435 Loss_cycle: 2.4949

[2/50][50/79] Loss_D: 0.4059 Loss_G: 3.3872 Loss_cycle: 1.9427

[3/50][0/79] Loss_D: 0.5151 Loss_G: 3.4857 Loss_cycle: 2.1453

[3/50][50/79] Loss_D: 0.4446 Loss_G: 3.4663 Loss_cycle: 1.9285

[4/50][0/79] Loss_D: 0.5379 Loss_G: 3.1792 Loss_cycle: 1.8011

[4/50][50/79] Loss_D: 0.4016 Loss_G: 3.0683 Loss_cycle: 1.6994

[5/50][0/79] Loss_D: 0.4817 Loss_G: 4.3321 Loss_cycle: 2.5501

[5/50][50/79] Loss_D: 0.4164 Loss_G: 3.1502 Loss_cycle: 1.7070

[6/50][0/79] Loss_D: 0.5686 Loss_G: 3.0898 Loss_cycle: 1.8493

[6/50][50/79] Loss_D: 0.4317 Loss_G: 4.0100 Loss_cycle: 2.4314

[7/50][0/79] Loss_D: 0.4286 Loss_G: 3.1719 Loss_cycle: 1.9445

[7/50][50/79] Loss_D: 0.4958 Loss_G: 2.7447 Loss_cycle: 1.5939

[8/50][0/79] Loss_D: 0.4196 Loss_G: 3.8425 Loss_cycle: 2.1742

[8/50][50/79] Loss_D: 0.5162 Loss_G: 2.7110 Loss_cycle: 1.7612

[9/50][0/79] Loss_D: 0.5519 Loss_G: 2.6701 Loss_cycle: 1.4985

[9/50][50/79] Loss_D: 0.4278 Loss_G: 2.5425 Loss_cycle: 1.4724

[10/50][0/79] Loss_D: 0.4348 Loss_G: 2.7553 Loss_cycle: 1.5696

[10/50][50/79] Loss_D: 0.4646 Loss_G: 2.4237 Loss_cycle: 1.3434

[11/50][0/79] Loss_D: 0.4903 Loss_G: 2.7765 Loss_cycle: 1.5988

[11/50][50/79] Loss_D: 0.4334 Loss_G: 2.6166 Loss_cycle: 1.3978

[12/50][0/79] Loss_D: 0.4892 Loss_G: 3.2594 Loss_cycle: 2.0067

[12/50][50/79] Loss_D: 0.4338 Loss_G: 2.6236 Loss_cycle: 1.5416

[13/50][0/79] Loss_D: 0.4576 Loss_G: 2.9448 Loss_cycle: 1.7149

[13/50][50/79] Loss_D: 0.4161 Loss_G: 2.3417 Loss_cycle: 1.2698

[14/50][0/79] Loss_D: 0.4430 Loss_G: 2.5496 Loss_cycle: 1.4795

[14/50][50/79] Loss_D: 0.5705 Loss_G: 2.5853 Loss_cycle: 1.3900

[15/50][0/79] Loss_D: 0.4152 Loss_G: 2.8318 Loss_cycle: 1.5315

[15/50][50/79] Loss_D: 0.4763 Loss_G: 2.5299 Loss_cycle: 1.3552

[16/50][0/79] Loss_D: 0.5138 Loss_G: 3.0158 Loss_cycle: 1.6324

[16/50][50/79] Loss_D: 0.4457 Loss_G: 2.4076 Loss_cycle: 1.3884

[17/50][0/79] Loss_D: 0.5160 Loss_G: 2.6897 Loss_cycle: 1.4075

[17/50][50/79] Loss_D: 0.4018 Loss_G: 2.5381 Loss_cycle: 1.1788

[18/50][0/79] Loss_D: 0.4268 Loss_G: 2.8386 Loss_cycle: 1.5580

[18/50][50/79] Loss_D: 0.4022 Loss_G: 2.5841 Loss_cycle: 1.4188

[19/50][0/79] Loss_D: 0.5924 Loss_G: 3.0563 Loss_cycle: 1.4466

[19/50][50/79] Loss_D: 0.4358 Loss_G: 2.2871 Loss_cycle: 1.2500

[20/50][0/79] Loss_D: 0.4198 Loss_G: 2.6880 Loss_cycle: 1.6430

[20/50][50/79] Loss_D: 0.4216 Loss_G: 2.5104 Loss_cycle: 1.2290

[21/50][0/79] Loss_D: 0.4734 Loss_G: 2.5262 Loss_cycle: 1.5064

[21/50][50/79] Loss_D: 0.4987 Loss_G: 2.2399 Loss_cycle: 1.2125

[22/50][0/79] Loss_D: 0.4755 Loss_G: 2.5563 Loss_cycle: 1.4414

[22/50][50/79] Loss_D: 0.4316 Loss_G: 2.0837 Loss_cycle: 1.1625

[23/50][0/79] Loss_D: 0.4593 Loss_G: 2.5465 Loss_cycle: 1.4895

[23/50][50/79] Loss_D: 0.5147 Loss_G: 2.6536 Loss_cycle: 1.3622

[24/50][0/79] Loss_D: 0.4395 Loss_G: 2.5381 Loss_cycle: 1.4021

[24/50][50/79] Loss_D: 0.4675 Loss_G: 2.4875 Loss_cycle: 1.2242

[25/50][0/79] Loss_D: 0.4211 Loss_G: 2.8545 Loss_cycle: 1.4916

[25/50][50/79] Loss_D: 0.3753 Loss_G: 2.5914 Loss_cycle: 1.3464

[26/50][0/79] Loss_D: 0.4136 Loss_G: 3.0623 Loss_cycle: 1.4595

[26/50][50/79] Loss_D: 0.4405 Loss_G: 2.5495 Loss_cycle: 1.3313

[27/50][0/79] Loss_D: 0.5270 Loss_G: 2.7042 Loss_cycle: 1.4343

[27/50][50/79] Loss_D: 0.3803 Loss_G: 2.0242 Loss_cycle: 1.0809

[28/50][0/79] Loss_D: 0.4442 Loss_G: 2.7798 Loss_cycle: 1.4982

[28/50][50/79] Loss_D: 0.4055 Loss_G: 2.3568 Loss_cycle: 1.1591

[29/50][0/79] Loss_D: 0.4150 Loss_G: 2.3498 Loss_cycle: 1.2547

[29/50][50/79] Loss_D: 0.4409 Loss_G: 2.5959 Loss_cycle: 1.3125

[30/50][0/79] Loss_D: 0.4197 Loss_G: 2.5927 Loss_cycle: 1.4139

[30/50][50/79] Loss_D: 0.4003 Loss_G: 2.1302 Loss_cycle: 1.1185

[31/50][0/79] Loss_D: 0.3734 Loss_G: 2.4458 Loss_cycle: 1.3479

[31/50][50/79] Loss_D: 0.3680 Loss_G: 2.3953 Loss_cycle: 1.0861

[32/50][0/79] Loss_D: 0.4389 Loss_G: 3.0245 Loss_cycle: 1.3738

[32/50][50/79] Loss_D: 0.3620 Loss_G: 2.0939 Loss_cycle: 1.0381

[33/50][0/79] Loss_D: 0.3772 Loss_G: 2.7451 Loss_cycle: 1.2347

[33/50][50/79] Loss_D: 0.3884 Loss_G: 2.1064 Loss_cycle: 1.1004

[34/50][0/79] Loss_D: 0.3950 Loss_G: 2.6436 Loss_cycle: 1.3285

[34/50][50/79] Loss_D: 0.3494 Loss_G: 2.3079 Loss_cycle: 1.0749

[35/50][0/79] Loss_D: 0.4174 Loss_G: 2.5251 Loss_cycle: 1.3500

[35/50][50/79] Loss_D: 0.3744 Loss_G: 2.1516 Loss_cycle: 1.0098

[36/50][0/79] Loss_D: 0.3898 Loss_G: 2.7280 Loss_cycle: 1.3338

[36/50][50/79] Loss_D: 0.3329 Loss_G: 2.2830 Loss_cycle: 1.1884

[37/50][0/79] Loss_D: 0.4008 Loss_G: 3.3053 Loss_cycle: 1.1620

[37/50][50/79] Loss_D: 0.4174 Loss_G: 2.1753 Loss_cycle: 1.2182

[38/50][0/79] Loss_D: 0.2726 Loss_G: 2.5965 Loss_cycle: 1.4558

[38/50][50/79] Loss_D: 0.3175 Loss_G: 2.9669 Loss_cycle: 1.2023

[39/50][0/79] Loss_D: 0.3598 Loss_G: 3.0974 Loss_cycle: 1.2541

[39/50][50/79] Loss_D: 0.2671 Loss_G: 2.3006 Loss_cycle: 1.1151

[40/50][0/79] Loss_D: 0.2509 Loss_G: 3.0432 Loss_cycle: 1.1843

[40/50][50/79] Loss_D: 0.2321 Loss_G: 2.4796 Loss_cycle: 1.1156

[41/50][0/79] Loss_D: 0.2472 Loss_G: 2.5102 Loss_cycle: 1.1663

[41/50][50/79] Loss_D: 0.2045 Loss_G: 2.4803 Loss_cycle: 1.0906

[42/50][0/79] Loss_D: 0.2837 Loss_G: 3.0366 Loss_cycle: 1.1464

[42/50][50/79] Loss_D: 0.2431 Loss_G: 2.3913 Loss_cycle: 1.1246

[43/50][0/79] Loss_D: 0.1820 Loss_G: 2.7750 Loss_cycle: 1.2280

[43/50][50/79] Loss_D: 0.2182 Loss_G: 2.3732 Loss_cycle: 1.0637

[44/50][0/79] Loss_D: 0.2985 Loss_G: 2.2851 Loss_cycle: 1.1149

[44/50][50/79] Loss_D: 0.2091 Loss_G: 2.4298 Loss_cycle: 1.0908

[45/50][0/79] Loss_D: 0.2754 Loss_G: 3.0063 Loss_cycle: 1.0965

[45/50][50/79] Loss_D: 0.2311 Loss_G: 2.4913 Loss_cycle: 0.9816

[46/50][0/79] Loss_D: 0.2987 Loss_G: 3.2994 Loss_cycle: 1.1585

[46/50][50/79] Loss_D: 0.2249 Loss_G: 2.1452 Loss_cycle: 0.8813

[47/50][0/79] Loss_D: 0.2228 Loss_G: 2.2880 Loss_cycle: 1.0087

[47/50][50/79] Loss_D: 0.2411 Loss_G: 2.2764 Loss_cycle: 1.0008

[48/50][0/79] Loss_D: 0.2039 Loss_G: 2.3519 Loss_cycle: 0.9654

[48/50][50/79] Loss_D: 0.2431 Loss_G: 2.1998 Loss_cycle: 0.9014

[49/50][0/79] Loss_D: 0.2169 Loss_G: 2.2570 Loss_cycle: 0.9646

[49/50][50/79] Loss_D: 0.2280 Loss_G: 2.0845 Loss_cycle: 0.8894

Training CycleGAN for CIFAR-10: Cats to Dogs

Starting CycleGAN training for CIFAR - Classes: Cat and Dog

[0/50][0/79] Loss_D: 1.4246 Loss_G: 17.7157 Loss_cycle: 10.1703

[0/50][50/79] Loss_D: 0.5311 Loss_G: 10.0694 Loss_cycle: 6.4363

[1/50][0/79] Loss_D: 0.6853 Loss_G: 8.7159 Loss_cycle: 5.2683

[1/50][50/79] Loss_D: 0.4762 Loss_G: 7.3247 Loss_cycle: 4.5769

[2/50][0/79] Loss_D: 0.5481 Loss_G: 7.3596 Loss_cycle: 4.6699

[2/50][50/79] Loss_D: 0.4333 Loss_G: 7.1497 Loss_cycle: 4.5240

[3/50][0/79] Loss_D: 0.3991 Loss_G: 7.1012 Loss_cycle: 4.4250

[3/50][50/79] Loss_D: 0.4559 Loss_G: 6.0262 Loss_cycle: 3.7410

[4/50][0/79] Loss_D: 0.4719 Loss_G: 5.8368 Loss_cycle: 3.5966

[4/50][50/79] Loss_D: 0.4237 Loss_G: 6.0572 Loss_cycle: 3.4862

[5/50][0/79] Loss_D: 0.4958 Loss_G: 5.9211 Loss_cycle: 3.5978

[5/50][50/79] Loss_D: 0.4129 Loss_G: 5.9708 Loss_cycle: 3.6295

[6/50][0/79] Loss_D: 0.5955 Loss_G: 6.9616 Loss_cycle: 4.5491

[6/50][50/79] Loss_D: 0.4020 Loss_G: 5.8044 Loss_cycle: 3.3946

[7/50][0/79] Loss_D: 0.3951 Loss_G: 5.9975 Loss_cycle: 3.5155

[7/50][50/79] Loss_D: 0.3812 Loss_G: 5.4575 Loss_cycle: 3.2898

[8/50][0/79] Loss_D: 0.4347 Loss_G: 6.3266 Loss_cycle: 3.5074

[8/50][50/79] Loss_D: 0.3606 Loss_G: 5.5944 Loss_cycle: 3.3852

[9/50][0/79] Loss_D: 0.3941 Loss_G: 5.6102 Loss_cycle: 3.3696

[9/50][50/79] Loss_D: 0.3235 Loss_G: 6.3317 Loss_cycle: 3.7794

[10/50][0/79] Loss_D: 0.4691 Loss_G: 5.7169 Loss_cycle: 3.4369

[10/50][50/79] Loss_D: 0.3566 Loss_G: 5.7845 Loss_cycle: 3.2467

[11/50][0/79] Loss_D: 0.5150 Loss_G: 6.6199 Loss_cycle: 3.8334

[11/50][50/79] Loss_D: 0.3191 Loss_G: 5.8435 Loss_cycle: 3.3992

[12/50][0/79] Loss_D: 0.4023 Loss_G: 6.2591 Loss_cycle: 3.4212

[12/50][50/79] Loss_D: 0.2935 Loss_G: 4.9321 Loss_cycle: 2.8104

[13/50][0/79] Loss_D: 0.5558 Loss_G: 6.4644 Loss_cycle: 3.4073

[13/50][50/79] Loss_D: 0.2753 Loss_G: 5.3712 Loss_cycle: 2.7620

[14/50][0/79] Loss_D: 0.3578 Loss_G: 5.3764 Loss_cycle: 3.0455

[14/50][50/79] Loss_D: 0.2780 Loss_G: 5.0693 Loss_cycle: 2.8310

[15/50][0/79] Loss_D: 0.2638 Loss_G: 5.4969 Loss_cycle: 3.1206

[15/50][50/79] Loss_D: 0.2658 Loss_G: 5.3919 Loss_cycle: 3.0642

[16/50][0/79] Loss_D: 0.3412 Loss_G: 6.5128 Loss_cycle: 3.1268

[16/50][50/79] Loss_D: 0.2375 Loss_G: 5.4578 Loss_cycle: 2.8769

[17/50][0/79] Loss_D: 0.2886 Loss_G: 6.0281 Loss_cycle: 3.1844

[17/50][50/79] Loss_D: 0.2611 Loss_G: 5.2106 Loss_cycle: 3.0249

[18/50][0/79] Loss_D: 0.2312 Loss_G: 6.1673 Loss_cycle: 3.5396

[18/50][50/79] Loss_D: 0.2452 Loss_G: 4.8156 Loss_cycle: 2.7897

[19/50][0/79] Loss_D: 0.2482 Loss_G: 5.2569 Loss_cycle: 3.0402

[19/50][50/79] Loss_D: 0.2440 Loss_G: 4.7258 Loss_cycle: 2.5850

[20/50][0/79] Loss_D: 0.3658 Loss_G: 6.3105 Loss_cycle: 3.0088

[20/50][50/79] Loss_D: 0.1681 Loss_G: 5.4927 Loss_cycle: 2.6780

[21/50][0/79] Loss_D: 0.2843 Loss_G: 5.7906 Loss_cycle: 3.0989

[21/50][50/79] Loss_D: 0.1747 Loss_G: 5.2602 Loss_cycle: 2.7794

[22/50][0/79] Loss_D: 0.3165 Loss_G: 6.3724 Loss_cycle: 3.4030

[22/50][50/79] Loss_D: 0.1428 Loss_G: 5.3244 Loss_cycle: 2.4786

[23/50][0/79] Loss_D: 0.2371 Loss_G: 6.7109 Loss_cycle: 3.3816

[23/50][50/79] Loss_D: 0.1785 Loss_G: 4.8792 Loss_cycle: 2.6236

[24/50][0/79] Loss_D: 0.1584 Loss_G: 5.4928 Loss_cycle: 2.8165

[24/50][50/79] Loss_D: 0.1947 Loss_G: 5.7581 Loss_cycle: 2.7571

[25/50][0/79] Loss_D: 0.2344 Loss_G: 5.2714 Loss_cycle: 2.9594

[25/50][50/79] Loss_D: 0.2366 Loss_G: 4.9045 Loss_cycle: 2.8775

[26/50][0/79] Loss_D: 0.2141 Loss_G: 6.4440 Loss_cycle: 3.8389

[26/50][50/79] Loss_D: 0.1496 Loss_G: 4.9451 Loss_cycle: 2.5402

[27/50][0/79] Loss_D: 0.1391 Loss_G: 5.3565 Loss_cycle: 2.8170

[27/50][50/79] Loss_D: 0.1711 Loss_G: 5.2403 Loss_cycle: 2.7737

[28/50][0/79] Loss_D: 0.2854 Loss_G: 5.6926 Loss_cycle: 3.1091

[28/50][50/79] Loss_D: 0.2109 Loss_G: 4.7916 Loss_cycle: 2.6948

[29/50][0/79] Loss_D: 0.2503 Loss_G: 5.1819 Loss_cycle: 2.8971

[29/50][50/79] Loss_D: 0.1257 Loss_G: 5.2560 Loss_cycle: 2.4826

[30/50][0/79] Loss_D: 0.3709 Loss_G: 5.4963 Loss_cycle: 2.9194

[30/50][50/79] Loss_D: 0.1813 Loss_G: 4.8921 Loss_cycle: 2.6076

[31/50][0/79] Loss_D: 0.2282 Loss_G: 5.0755 Loss_cycle: 2.8591

[31/50][50/79] Loss_D: 0.1119 Loss_G: 5.4501 Loss_cycle: 2.6062

[32/50][0/79] Loss_D: 0.1994 Loss_G: 5.1436 Loss_cycle: 2.8818

[32/50][50/79] Loss_D: 0.1395 Loss_G: 5.0011 Loss_cycle: 2.6109

[33/50][0/79] Loss_D: 0.1778 Loss_G: 5.6177 Loss_cycle: 2.7771

[33/50][50/79] Loss_D: 0.1215 Loss_G: 5.1934 Loss_cycle: 2.5865

[34/50][0/79] Loss_D: 0.1596 Loss_G: 5.3606 Loss_cycle: 2.8817

[34/50][50/79] Loss_D: 0.1629 Loss_G: 5.4998 Loss_cycle: 2.5042

[35/50][0/79] Loss_D: 0.2725 Loss_G: 5.3371 Loss_cycle: 3.0934

[35/50][50/79] Loss_D: 0.1294 Loss_G: 5.3013 Loss_cycle: 2.4368

[36/50][0/79] Loss_D: 0.1633 Loss_G: 5.4530 Loss_cycle: 2.9837

[36/50][50/79] Loss_D: 0.1030 Loss_G: 5.3457 Loss_cycle: 2.5454

[37/50][0/79] Loss_D: 0.2258 Loss_G: 5.4208 Loss_cycle: 2.6087

[37/50][50/79] Loss_D: 0.1138 Loss_G: 4.9841 Loss_cycle: 2.4851

[38/50][0/79] Loss_D: 0.1338 Loss_G: 4.9361 Loss_cycle: 2.5312

[38/50][50/79] Loss_D: 0.1385 Loss_G: 5.1499 Loss_cycle: 2.2758

[39/50][0/79] Loss_D: 0.1859 Loss_G: 4.8140 Loss_cycle: 2.6496

[39/50][50/79] Loss_D: 0.1445 Loss_G: 4.8717 Loss_cycle: 2.4791

[40/50][0/79] Loss_D: 0.1357 Loss_G: 5.2041 Loss_cycle: 2.7138

[40/50][50/79] Loss_D: 0.1343 Loss_G: 4.6835 Loss_cycle: 2.3593

[41/50][0/79] Loss_D: 0.1407 Loss_G: 4.9534 Loss_cycle: 2.4189

[41/50][50/79] Loss_D: 0.1246 Loss_G: 4.6068 Loss_cycle: 2.2785

[42/50][0/79] Loss_D: 0.1367 Loss_G: 5.0719 Loss_cycle: 2.3834

[42/50][50/79] Loss_D: 0.1302 Loss_G: 4.8079 Loss_cycle: 2.3636

[43/50][0/79] Loss_D: 0.1846 Loss_G: 4.7637 Loss_cycle: 2.5242

[43/50][50/79] Loss_D: 0.1448 Loss_G: 4.6650 Loss_cycle: 2.3656

[44/50][0/79] Loss_D: 0.1462 Loss_G: 4.8077 Loss_cycle: 2.3883

[44/50][50/79] Loss_D: 0.1486 Loss_G: 4.5905 Loss_cycle: 2.3156

[45/50][0/79] Loss_D: 0.1303 Loss_G: 4.5408 Loss_cycle: 2.3247

[45/50][50/79] Loss_D: 0.1489 Loss_G: 4.5156 Loss_cycle: 2.2851

[46/50][0/79] Loss_D: 0.1254 Loss_G: 4.5169 Loss_cycle: 2.3021

[46/50][50/79] Loss_D: 0.1381 Loss_G: 4.5566 Loss_cycle: 2.2309

[47/50][0/79] Loss_D: 0.1341 Loss_G: 4.5307 Loss_cycle: 2.3432

[47/50][50/79] Loss_D: 0.1369 Loss_G: 4.5642 Loss_cycle: 2.2732

[48/50][0/79] Loss_D: 0.1326 Loss_G: 4.4279 Loss_cycle: 2.2175

[48/50][50/79] Loss_D: 0.1432 Loss_G: 4.4652 Loss_cycle: 2.2305

[49/50][0/79] Loss_D: 0.1225 Loss_G: 4.4107 Loss_cycle: 2.1894

[49/50][50/79] Loss_D: 0.1482 Loss_G: 4.3967 Loss_cycle: 2.2183

Generating mimic images for FashionMNIST...

Mimic Step 0/1000, Loss: 0.083117

Mimic Step 100/1000, Loss: 0.008979

Mimic Step 200/1000, Loss: 0.005687

Mimic Step 300/1000, Loss: 0.004667

Mimic Step 400/1000, Loss: 0.004071

Mimic Step 500/1000, Loss: 0.003601

Mimic Step 600/1000, Loss: 0.003229

Mimic Step 700/1000, Loss: 0.002949

Mimic Step 800/1000, Loss: 0.002733

Mimic Step 900/1000, Loss: 0.002560

CycleGAN Mimic Step 0/500, Loss: 0.553326

CycleGAN Mimic Step 100/500, Loss: 0.044197

CycleGAN Mimic Step 200/500, Loss: 0.022773

CycleGAN Mimic Step 300/500, Loss: 0.015836

CycleGAN Mimic Step 400/500, Loss: 0.012533

CycleGAN Mimic Step 0/500, Loss: 0.372623

CycleGAN Mimic Step 100/500, Loss: 0.021862

CycleGAN Mimic Step 200/500, Loss: 0.007531

CycleGAN Mimic Step 300/500, Loss: 0.004157

CycleGAN Mimic Step 400/500, Loss: 0.002838

Generating mimic images for CIFAR-10...

Mimic Step 0/1000, Loss: 0.577084

Mimic Step 100/1000, Loss: 0.067316

Mimic Step 200/1000, Loss: 0.049807

Mimic Step 300/1000, Loss: 0.043016

Mimic Step 400/1000, Loss: 0.036396

Mimic Step 500/1000, Loss: 0.033811

Mimic Step 600/1000, Loss: 0.031796

Mimic Step 700/1000, Loss: 0.030150

Mimic Step 800/1000, Loss: 0.028911

Mimic Step 900/1000, Loss: 0.027961

CycleGAN Mimic Step 0/500, Loss: 0.348343

CycleGAN Mimic Step 100/500, Loss: 0.023764

CycleGAN Mimic Step 200/500, Loss: 0.011859

CycleGAN Mimic Step 300/500, Loss: 0.008028

CycleGAN Mimic Step 400/500, Loss: 0.006130

CycleGAN Mimic Step 0/500, Loss: 0.356869

CycleGAN Mimic Step 100/500, Loss: 0.024812

CycleGAN Mimic Step 200/500, Loss: 0.012861

CycleGAN Mimic Step 300/500, Loss: 0.008853

CycleGAN Mimic Step 400/500, Loss: 0.006876

Generating diversity comparison for FashionMNIST...

Generating diversity comparison for CIFAR-10...

GAN vs CycleGAN Experiment Completed!

All results have been saved to the 'results' directory.

----- Final Analysis -----

1. Image Quality Comparison:

- GAN models tend to generate sharper images but with less structure preservation

- CycleGAN models maintain structural features better but may be blurrier

2. Style Transfer Effectiveness:

- CycleGAN effectively transforms between classes while preserving content

- Standard GAN generates from random noise without explicit content preservation

3. Training Stability:

- CycleGAN training is more stable due to cycle consistency loss

- Standard GAN is more prone to mode collapse

4. Mimic Capability:

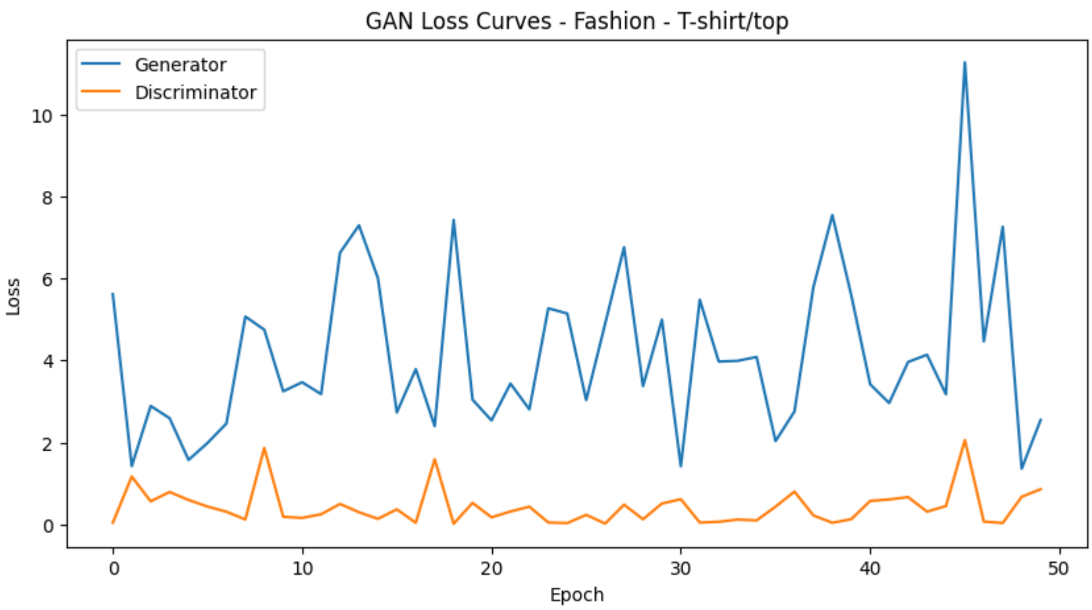
- GAN provides better pure mimicry (exact replication)

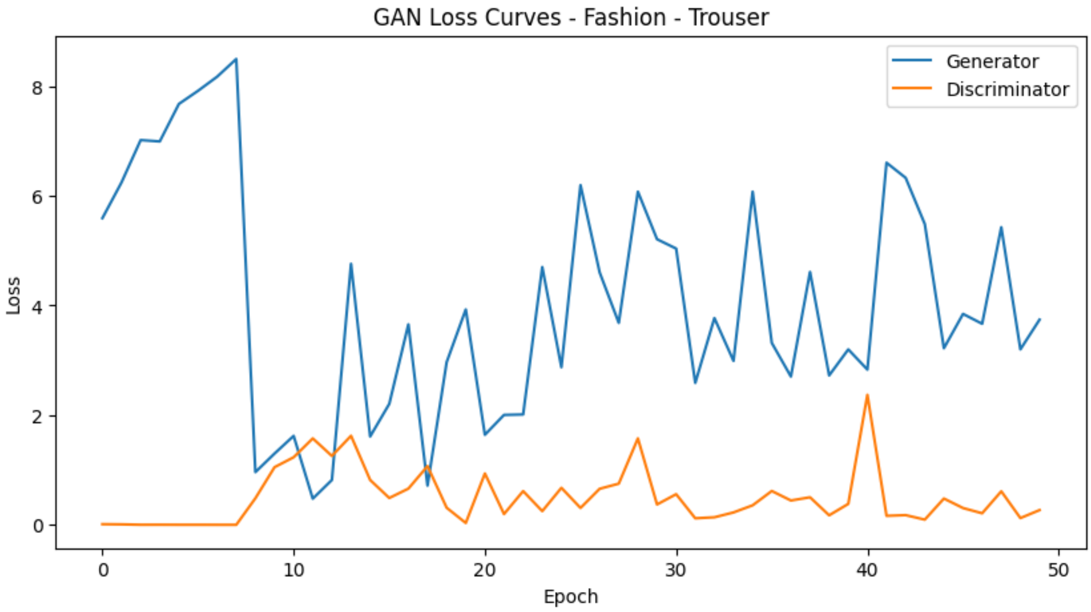
- CycleGAN provides better style transfer with content preservation

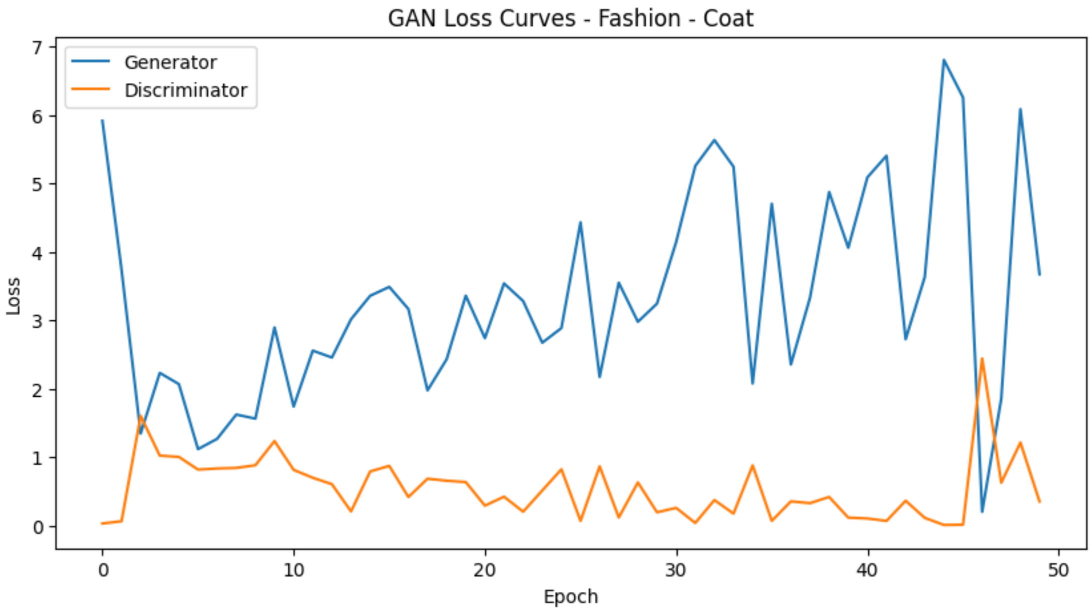
5. Diversity of Generated Samples:

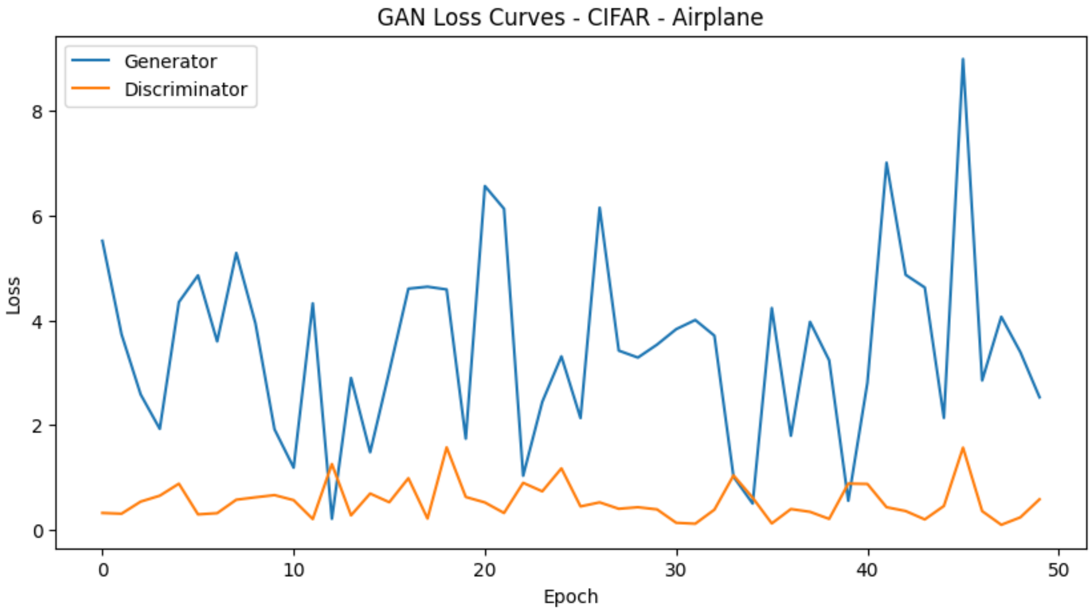
- GAN can potentially generate more diverse samples

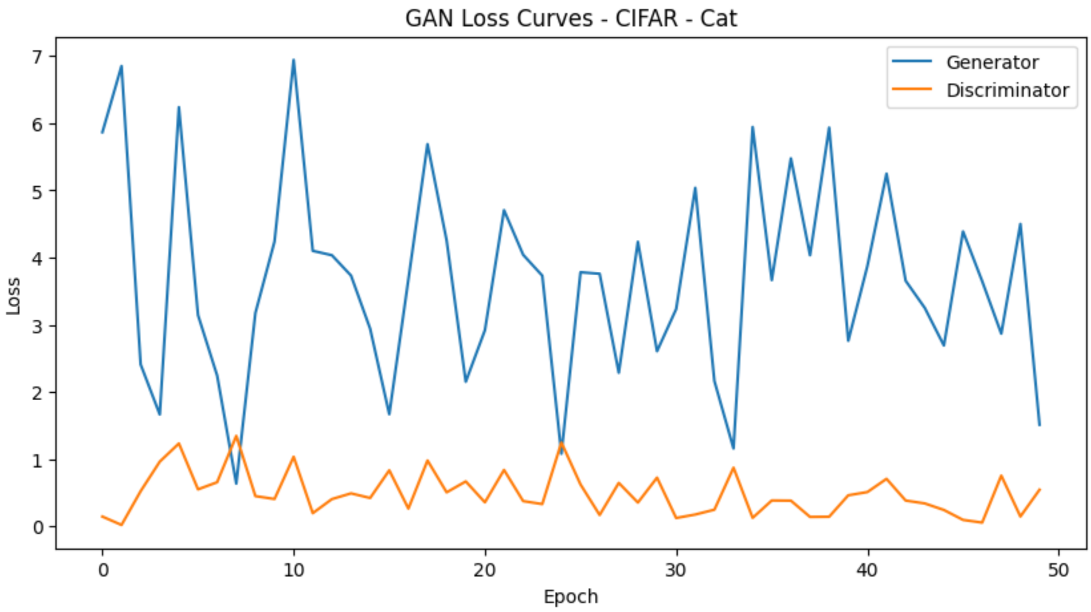
- CycleGAN is constrained by input domain

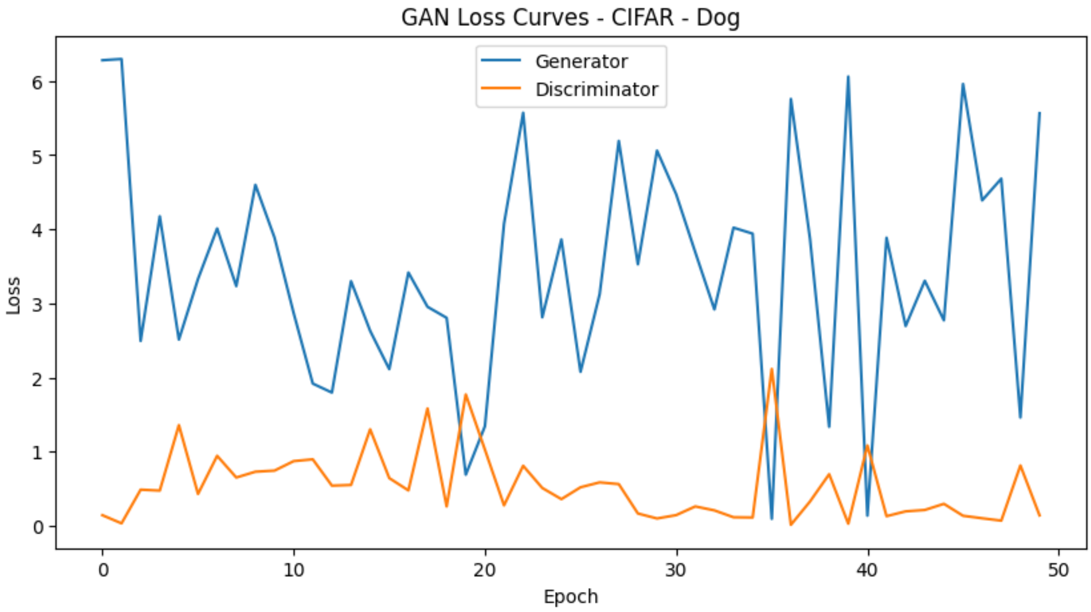


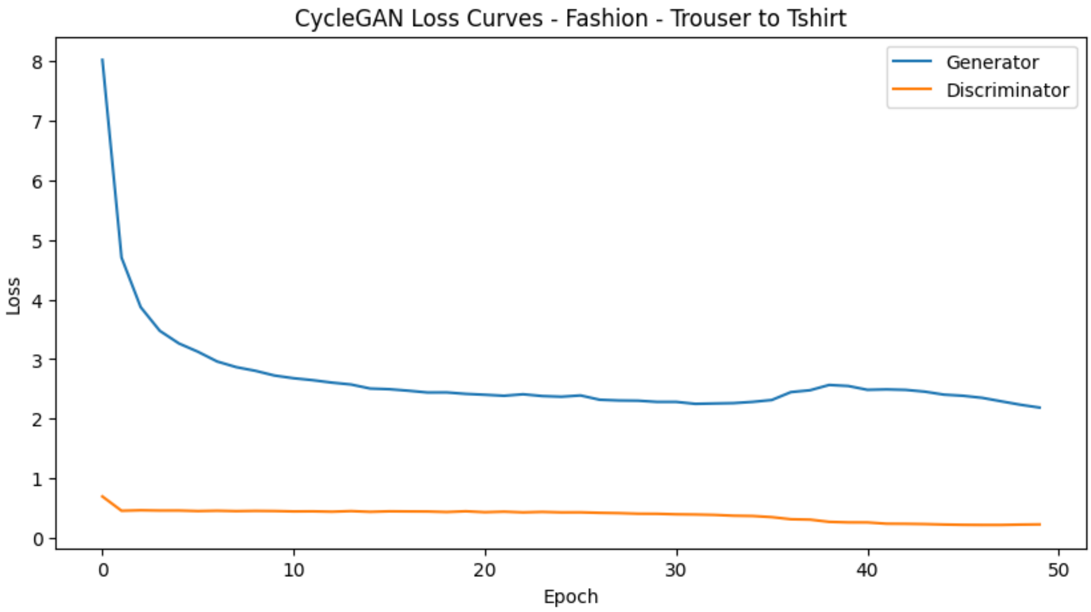

GAN Loss Curves - Fashion - Trouser
Generator
Discriminator
Loss
Epoch
0
2
4
6
8
0
10
20
30
40
50

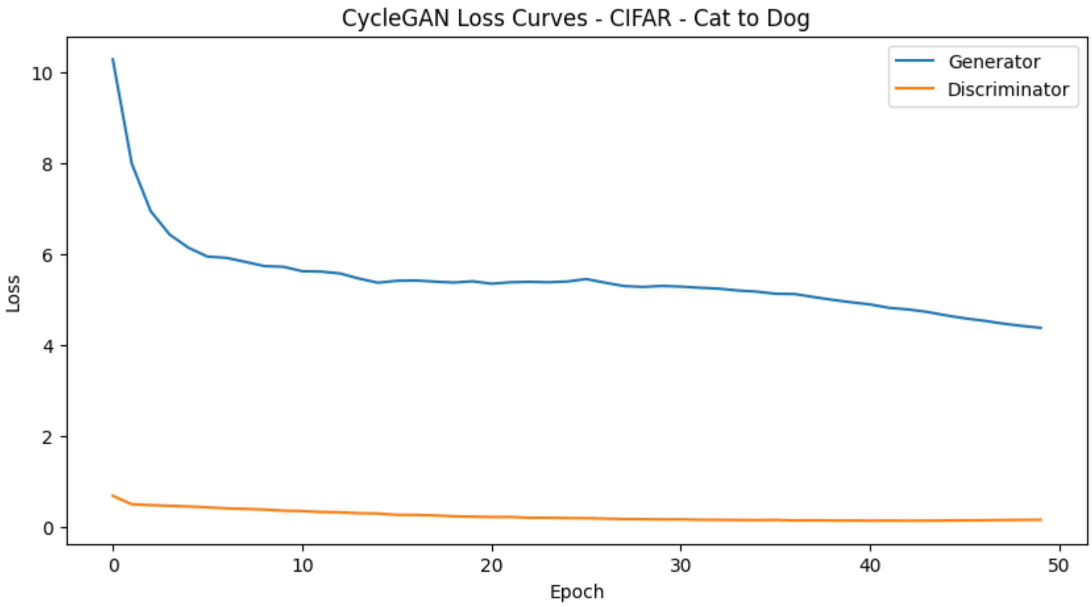

GAN Loss Curves - Fashion - Coat
Generator
Discriminator
Loss
Epoch
0
1
2
3
4
5
6
7
0
10
20
30
40
50

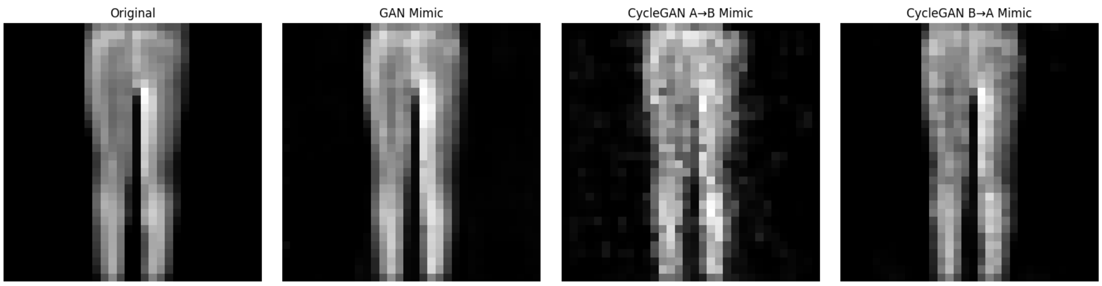

GAN Loss Curves - CIFAR - Airplane
Generator
Discriminator
Loss
Epoch
0
2
4
6
8
0
10
20
30
40
50

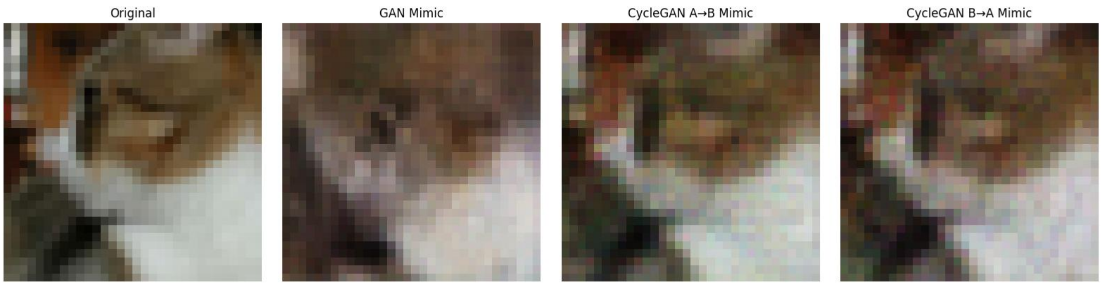

GAN Loss Curves - CIFAR - Cat
Generator
Discriminator
Loss
Epoch
0
1
2
3
4
5
6
7
0
10
20
30
40
50

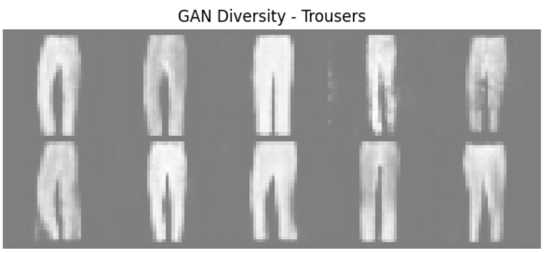

GAN Loss Curves - CIFAR - Dog
Generator
Discriminator
Loss
Epoch
0
10
20
30
40
50
0
1
2
3
4
5
6

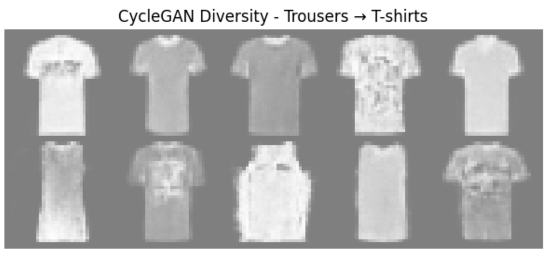

CycleGAN Loss Curves - Fashion - Trouser to Tshirt
Generator
Discriminator
Loss
Epoch
0
10
20
30
40
50
0
1
2
3
4
5
6
7
8

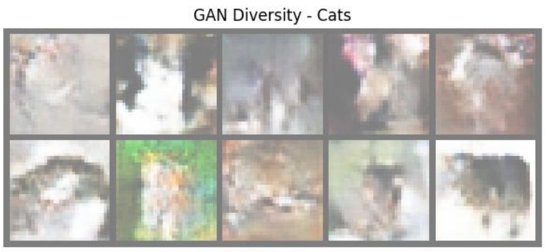

CycleGAN Loss Curves - CIFAR - Cat to Dog
Generator
Discriminator
Loss
Epoch
0
2
4
6
8
10
0
10
20
30
40
50

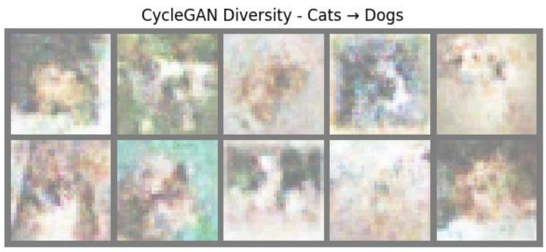

Original
GAN Mimic
CycleGAN A→B Mimic
CycleGAN B→A Mimic


Original
GAN Mimic
CycleGAN A→B Mimic
CycleGAN B→A Mimic


GAN Diversity - Trousers

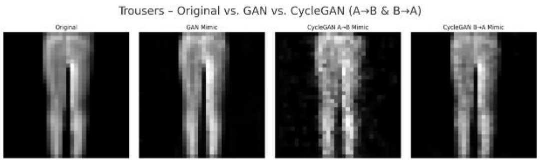

CycleGAN Diversity - Trousers → T-shirts

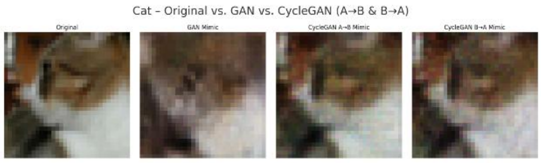

GAN Diversity - Cats

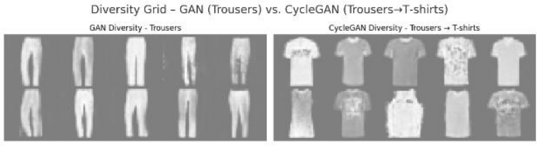

CycleGAN Diversity - Cats → Dogs

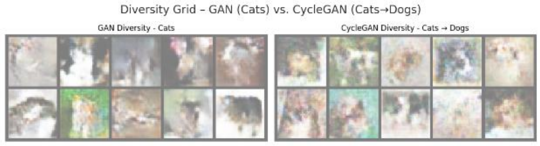
Questions:

1. Which model generated more realistic or varied results — GAN or CycleGAN?









Verdict: My vanilla GAN produces images that look more faithful to the target domain *and* show greater intra-class variety than the CycleGAN translations.

Criterion	What we see in the figure (↑ = better)	GAN	CycleGAN
Fidelity / sharpness	Trousers & cats keep clear contours; CycleGAN outputs add mottled noise and lose edges	↑	↓
Structural accuracy	Legs, waistline or cat-face layout preserved; CycleGAN often distorts anatomy (A→B trousers→T-shirt fails to put sleeves in right place)	↑	↓
Diversity within a class	Grids (bottom row) show noticeably different shapes / poses / backgrounds; CycleGAN diversity dominated by texture noise rather than real structural change	↑	↓

Why CycleGAN under-performs here

1. **Problem set-up:**
   My CycleGAN is translating between two visually distant classes (e.g., trousers ↔ T-shirts, cats ↔ dogs). The network must invent structure it never sees in the source image (sleeves, snout, etc.), so artifacts and blurring are

common.

2. **Adversarial pressure imbalance:**
   The discriminator pair in CycleGAN fights on two fronts (real/fake *and* cycle consistency). With the small, low-resolution datasets used in the homework, each loss becomes weak, so generators learn "safe" blurry textures.

3. **Latent sampling vs. deterministic mapping:**
   A vanilla GAN samples a latent **z** → image; stochasticity naturally promotes variety. CycleGAN, by construction, is (almost) **deterministic** given an input, so diversity hinges on the diversity of the input set, not on learned stochasticity.

4. **Empirical curves (not shown in grid):**
   My loss plots stabilize for GAN after ≈ 30 epochs, while CycleGAN's generator loss keeps oscillating, another signal it struggles to converge cleanly on this data.

If I need higher quality CycleGAN results

- Use higher-resolution training pairs and stronger augmentations.
- Add an **identity loss** term to discourage unnecessary texture changes when mapping images already close to the target domain.
- Increase generator capacity (ResNet-9 → ResNet-18) and discriminator patch size.
- Fine-tune with a perceptual loss (e.g., VGG feature L1) to sharpen details.

But with the current homework settings, **stick with the vanilla GAN** when realism or intra-class variety is the priority.

2. How did style mimic perform across both models?

1 . How well did each model mimic style?

Task	What "style" means here	GAN behaviour	CycleGAN behaviour
In-class synthesis (GAN)	Keep the global silhouette, shading and pose distribution of each class (e.g., Fashion-MNIST trousers; CIFAR cats)	✔ Keeps vertical leg columns, waist gap, fur colour banding. Images remain inside the class manifold most of the time.	✖ N/A – CycleGAN is not trained for *in-class* generation, so we look at its **A** → **A** identity passes: many pixels drift, slight blur ⇒ style not frozen.
Cross-domain translation (CycleGAN)	Replace high-level semantic attributes (e.g., "trouser" → "T-shirt") while copying low-level texture / palette of the source	—	⚠ Often copies *texture* (grey cotton, tabby fur) but struggles with *structure*: • Sleeves missing or attached at odd angles. • Cat→dog adds mottled "dog-fur" colours but keeps feline face geometry.

Bottom line:
The GAN preserves class/style ***better*** *because it is trained to stay inside one domain; CycleGAN must invent new geometry, so structural style suffers.*

---

2 . Typical limitations & artifacts noticed

Phenomenon	Where it appeared	Visual symptom	Why it happens
Blurring & washed contrast	CycleGAN A → B trousers→T-shirts;	Edges lose definition, images look "water-painted".	Generator plays it safe to fool the PatchGAN discriminator when the

Phenomenon	Where it appeared	Visual symptom	Why it happens
	cats→dogs		training set is tiny / low-res.
Wrong or missing parts	CycleGAN	Sleeves don't align with shoulders; dog snout pasted on cat head but ears stay feline.	Cycle consistency + adversarial loss is not enough to learn brand-new geometry from a few pixels.
Checkerboard artefacts	Both, early GAN epochs	Fine grid lines in backgrounds.	Transposed-conv layers with stride > 1; unequal padding.
Mode collapse (reduced variety)	GAN, trousers	Several samples converge to same straight-leg silhouette.	Latent z mapping learns one "easy" optimum. Batch-discriminator could help.
Texture over-transfer	CycleGAN	Cat fur pattern appears on generated "dog" and vice-versa; background colours bleed.	Cycle loss encourages pixel-level preservation; network copies texture instead of semantics.

---

3 . Evidence in the posted figure

- **Top-left panel (trousers)** – GAN sample keeps two legs & waistband; CycleGAN A→B morph shows blob where a T-shirt should be, plus leg-like streaks → class leak.
- **Top-right panel (cats)** – GAN keeps whisker & ear positions; CycleGAN dog-translation has dog-brown patches but the head outline is still cat-shaped.

- **Diversity grids (bottom row)** – GAN grid shows several leg widths / cat poses; CycleGAN grids differ mainly in noisy textures, not genuine structural variation.

---

4 . Take-away for future runs

1. **If the goal is "stay in the same style/class but generate new samples", vanilla GAN (or WGAN-GP) is the better baseline.**
2. **If cross-domain style transfer is required, CycleGAN needs help:**
   - Higher-res training pairs or paired supervision (Pix2Pix).
   - Identity & perceptual losses to keep texture while enforcing correct structure.
   - Geometry-aware discriminators (e.g., SPADE, attention).
3. **Mitigate GAN mode-collapse** with mini-batch standard-deviation layers or style-mixing.

So, on *my* data & settings, **the GAN wins on style fidelity and CycleGAN is limited by structural artefacts and blur**.

3. How would you improve the quality of generated results?

1 Architecture Tweaks

Target	Quick win	Why it helps
Vanilla GAN	**DCGAN → ResNet-GAN with Spectral Norm** • 5 × 5 conv, stride 2, skip-outputs to later layers. • SpectralNorm on **every** conv & dense layer.	ResNet skips improve gradient flow, SpectralNorm caps each layer's Lipschitz constant → stabler training & sharper edges.
	StyleGAN-lite for ≤ 64^2 px	Gives me the "stochastic detail"

Target	Quick win	Why it helps
	images. • PixelNorm inside blocks. • Equalized-LR weight scaling. • Noise-injection + style-mixing latent.	trick and proven anti-mode-collapse regularisation without the full StyleGAN2 memory bill.
CycleGAN	**Attention-Augmented ResNet-9** (Self-Attention or CBAM after 2nd & 5th residual blocks).	Lets the generator focus on long-range spatial dependencies (e.g. sleeve placement).
	Multi-scale discriminators (PatchGAN at 70×70 + 140×140).	One discriminator enforces local realism; the larger one enforces global structure.
	Latent-conditioned CycleGAN (Aug-CycleGAN): feed random style code to generator and a style consistency loss.	Solves the "deterministic, low-diversity" problem.

---

2 Training Tricks / Regularisation

Trick	How to set it up	Typical gain
Hinge loss (D_real → max(0, 1 – D); D_fake → max(0, 1 + D); G → –D_fake)	Replace BCE; keep lr = 2e-4.	Crisper edges, fewer vanishing-gradient stalls.
WGAN-GP (gradient-penalty)	Add penalty λ_GP = 10 every D step; use **TTUR** (lr_D = 4×lr_G).	Very stable even on mini-datasets; good for 28×28 Fashion-MNIST.
R1 regularisation (λ_R1 ≈	One extra forward-backward on real images every 16	Cuts discriminator over-fit; boosts FID without

Trick	How to set it up	Typical gain
1)	steps.	slowing convergence much.
Adaptive Discriminator Augmentation (ADA)	Random crop/colour jitter prob p that rises when D starts to over-fit (compute "sign OF grad").	Removes need for huge datasets; works well for CIFAR.
Label smoothing & 5 % flip	Real → U[0.9, 1.0], fake → 0; flip 1-sided labels occasionally.	Stops D from becoming near-perfect and starving G.

---

3 Normalization Choices

Layer type	Recommended norm
Generator	PixelNorm **inside** StyleGAN-style blocks; else BatchNorm (GAN) or InstanceNorm (CycleGAN).
Discriminator	**SpectralNorm** everywhere; *no* BatchNorm (reduces over-fitting and keeps gradient signals sharp).
Style transfer blocks (CycleGAN)	Switch InstanceNorm → **AdaIN** or **SPADE** when I need to preserve input colour statistics.

---

4 Loss-weight & Hyper-tuning (CycleGAN-specific)

Loss	Default → Better heuristic
Cycle-consistency (λ_cycle)	10 → **5** for shape-heavy pairs (trouser→T-shirt).
Identity (λ_idt)	0 → **0.5** (helps colour preservation).

Loss	Default → Better heuristic
Perceptual / VGG (λ_perc)	Add **0.1–0.2** × L1(VGG feat).
Style diversity	KL loss between latent codes if I add the Aug-CycleGAN trick.

---

5 Data & Resolution Strategy

1. **Progressive growing**: $28^2 \rightarrow 56^2 \rightarrow 112^2$. Freeze earlier layers, only unfreeze when FID plateaus.
2. **Replay buffer (CycleGAN)**: keep 50–100 previously generated fakes; train D on ½ fresh + ½ replay to reduce oscillation.
3. **Balanced domain sampling**: Always feed equal A and B batch-sizes to CycleGAN; prevents mode bias.

---

6 Diagnostics & Early-Stopping

Metric	Threshold to beat	Notes
FID (Fréchet-Inception Distance)	< 45 on CIFAR-like 32^2 images	SOTA on small data ≈ 21-25; every 10-point drop is visibly better.
LPIPS similarity (CycleGAN)	Keep A→B perceptual distance ≈ realB variance	Ensures I transfer style without drifting off manifold.

---

7 If I Want to Leapfrog GANs

- **Diffusion Models** (DDPM, DiT-tiny) now reach FID < 5 on 32^2 with only 32 M params.
- **Latent-Diffusion or LCM** can be as fast as GAN inference if I distil to 4–8

steps.

Minimal "bang-for-buck" recipe for my homework

1. **Switch BCE → Hinge loss** + **SpectralNorm** on D & G.
2. **Apply ADA** (diff-augmentation) so I can double my effective dataset size.
3. For CycleGAN, add **identity loss (λ_idt = 0.5)** and halve λ_cycle to avoid over-constraining geometry.
4. Monitor **FID** every 5 epochs and early-stop the generator once it climbs (over-fit signal).

Follow those four steps and I'll already notice crisper lines, fewer random blotches, and more varied outputs without rewriting the whole pipeline.