

# Sketch2Pix: Generating Realistic Images from Sketches using Conditional GAN

Drishtant Jain  
24M1085

Seasons of Code 2025 at IIT Bombay

August 2, 2025

# Problem Statement

- Can we generate realistic face images from hand-drawn sketches?
- This problem is essential in:
  - Forensic sketch matching
  - Art and animation
  - Facial composite generation

# Proposed Solution

- We use a Conditional GAN (Pix2Pix) model
- Generator translates sketches to realistic face images
- Discriminator evaluates if image pairs (sketch, photo) are real or fake

# Dataset Overview

- Dataset: CUHK Face Sketch Database (CUFS)
- Contains aligned photo-sketch pairs
- Preprocessing:
  - Resize, crop, normalize
  - Train/Validation Split: 80% / 20%

# Generator Architecture

- U-Net architecture with skip connections
- Downsampling with Conv2D + LeakyReLU
- Upsampling with TransposedConv2D + BatchNorm + ReLU
- Output: Tanh-activated RGB image

# Discriminator Architecture

- PatchGAN discriminator
- Input: Concatenated sketch and photo
- Output: Real or fake prediction for each patch
- Promotes high-frequency realism

# Training Details

- Loss Functions:
  - Adversarial Loss: BCEWithLogitsLoss
  - Pixel-wise Loss: L1 Loss
- Optimization:
  - Adam Optimizer
  - Cosine Annealing Scheduler for Generator
  - Label Smoothing for stability

# Training Pipeline

- Load dataset using PyTorch 'Dataset' class
- Alternate updates for Generator and Discriminator
- Save images and checkpoints every 10 epochs
- Optimized for Kaggle GPU environment

# Results:

Sample: epoch\_90\_sample\_2.png



Sample: epoch\_90\_sample\_3.png



# Results:

==== Training Results ====

Sample: epoch\_90\_sample\_0.png



Sample: epoch\_90\_sample\_1.png



# Results (Before & After)

- Output improves as training progresses
- Sample results saved using ‘`save_image()`’ during training



## Results (Before hyperparameter tuning)

- Output improves as training progresses
- Sample results saved using ‘`save_image()`’ during training



## Results (Before before hyperparameter tuning)

- Output improves as training progresses
- Sample results saved using ‘`save_image()`’ during training

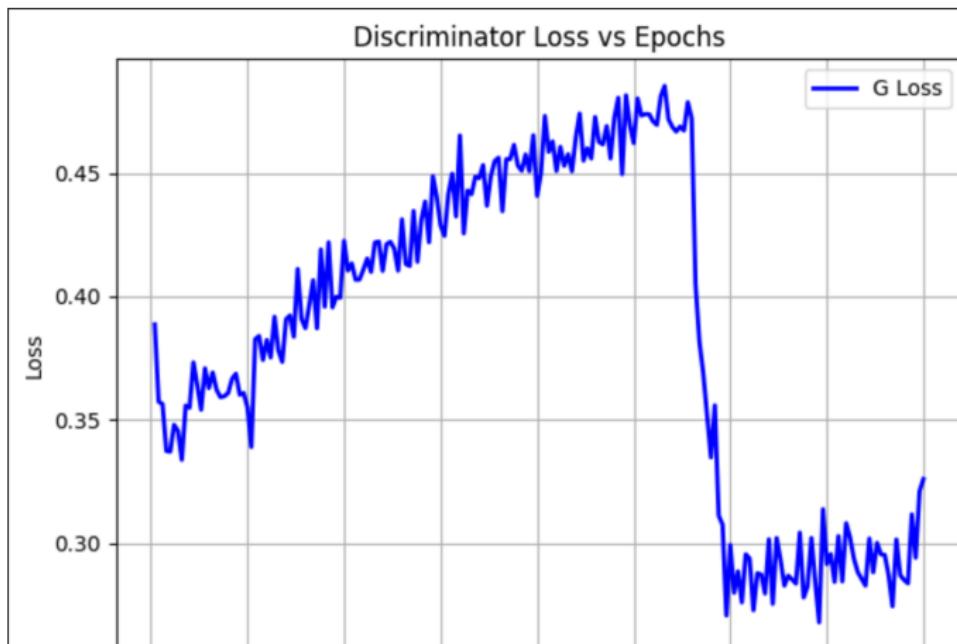


# Challenges and Improvements

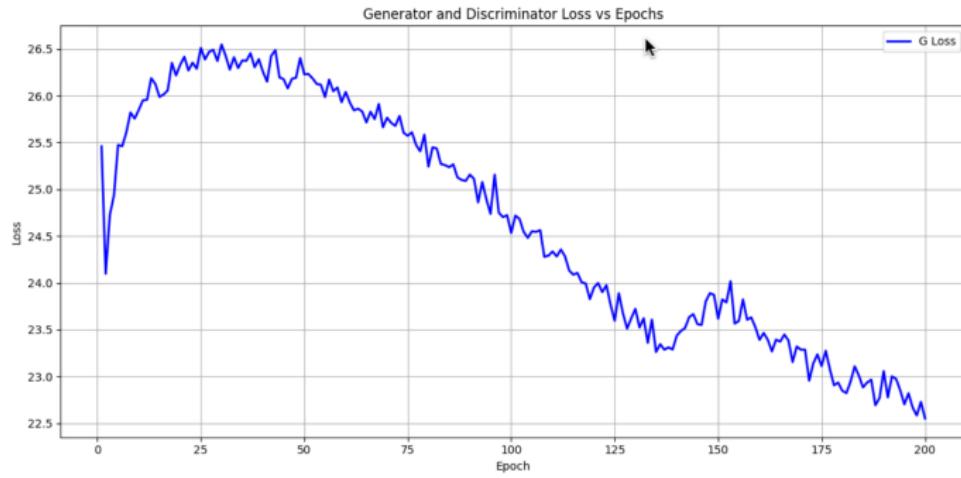
- Challenges:
  - Small dataset
  - Sketches have limited detail
- Future Improvements:
  - Use perceptual loss (VGG)
  - Try attention-based generators
  - Train on larger datasets

# Quantitative Results

- **Validation Dataset Size:** 296 matched pairs
- **Average L1 Loss on Validation Set:** 0.7219
- Generator and Discriminator loss curves over 200 epochs



# Quantitative Results



# Conclusion

- Trained a Pix2Pix model to convert sketches into realistic images
- Used CUFS dataset and custom PyTorch pipeline
- Results demonstrate photo reconstruction from edges

# Thank You

GitHub Repo