

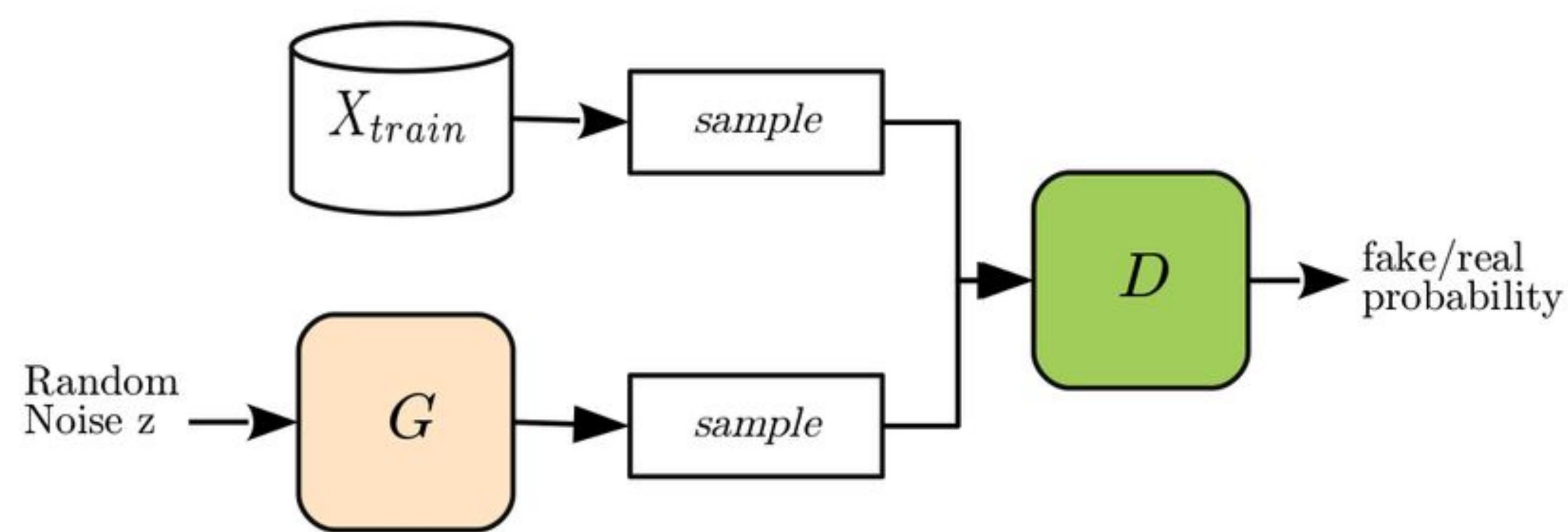
PROBLEM

- Evaluation of Generative Adversarial Network (GAN) on generating Anime Faces
- Based on project by Keras-AnimeFace GitHub repository, using the animeface-character-dataset
- Survey of different GAN architectures and parameters



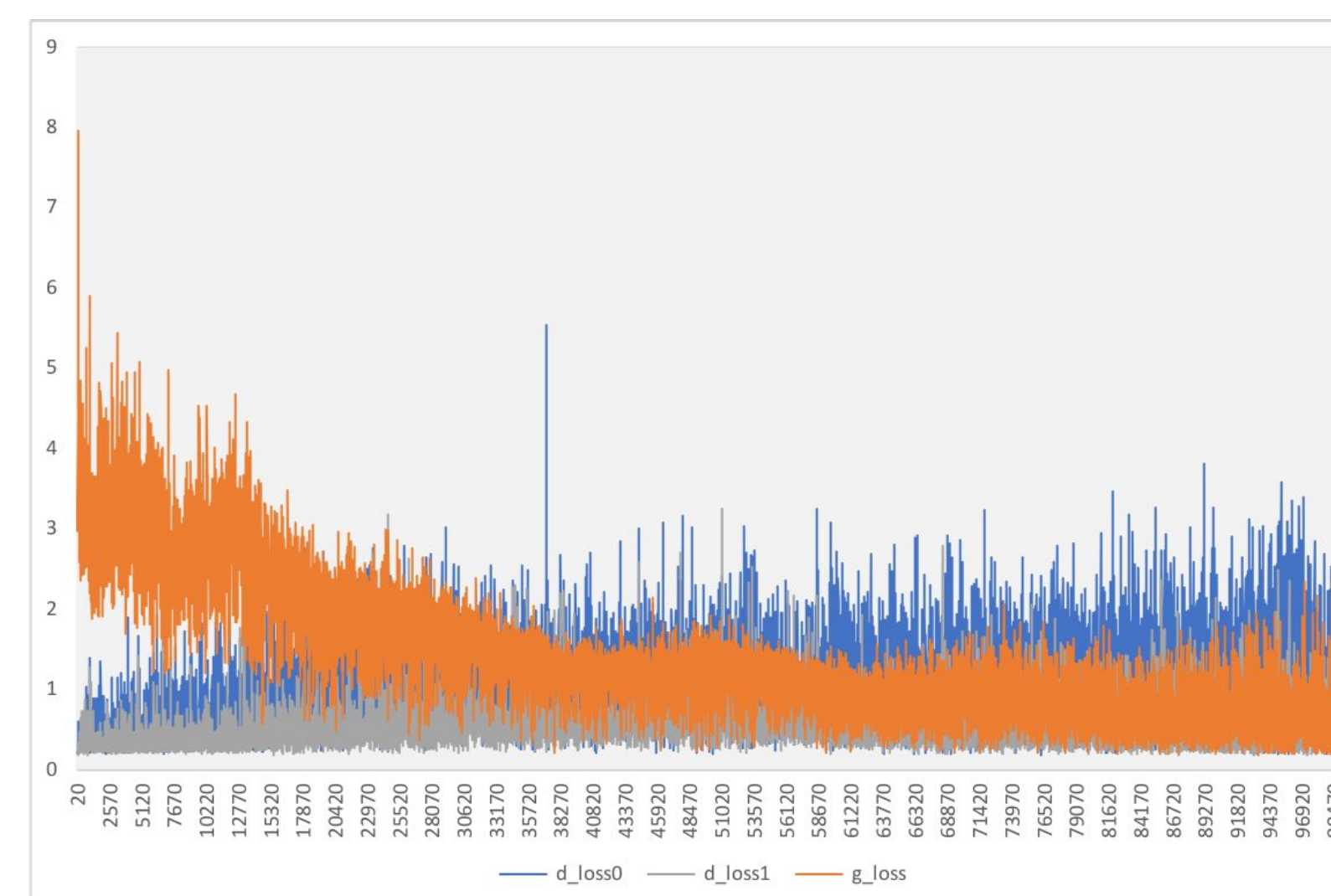
MODEL FORMULATION

- Generator (G) and Discriminator (D)
- G generates images to trick D
- D tries to discern between real and fake images
- Learn from each other: Zero Sum Game

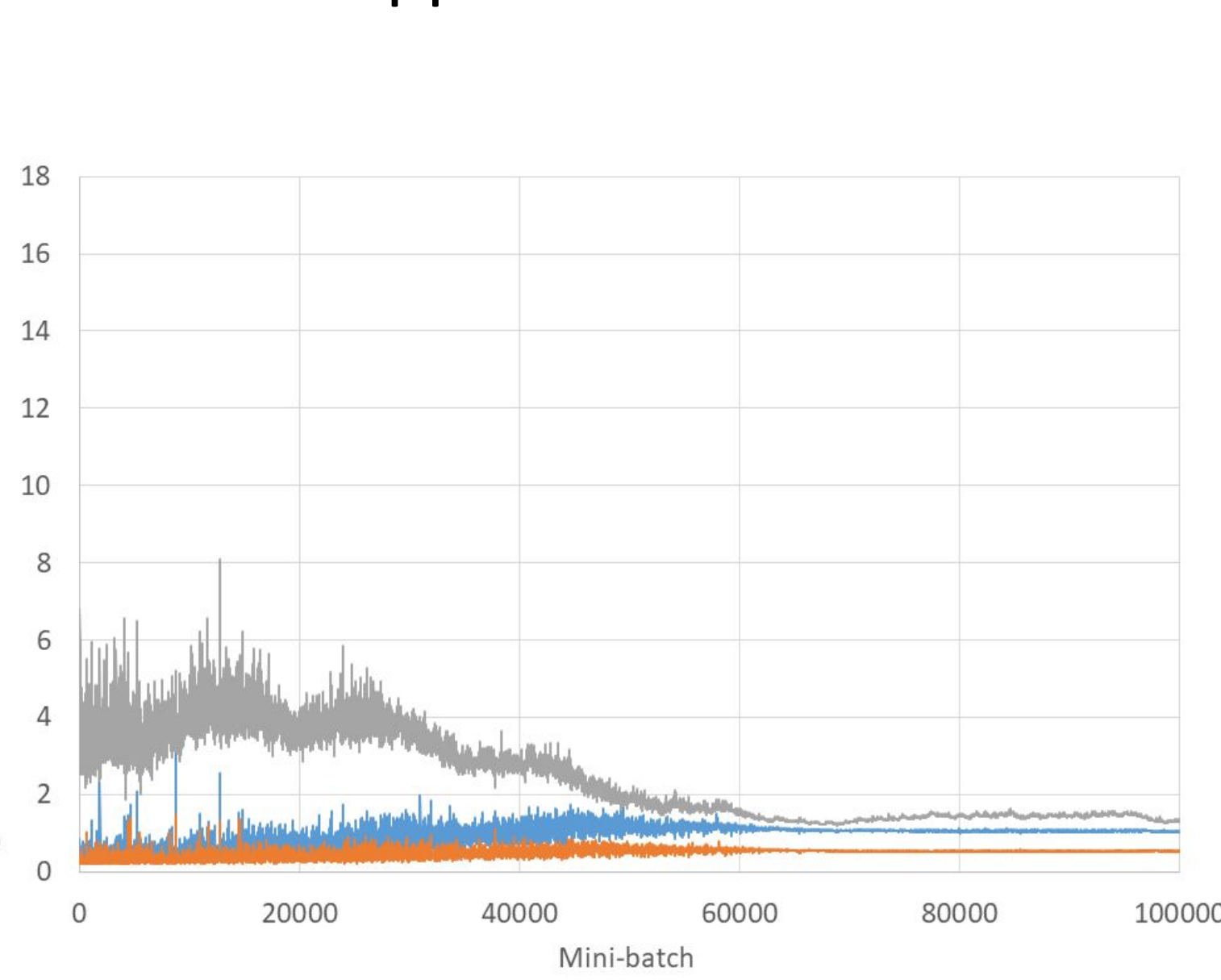
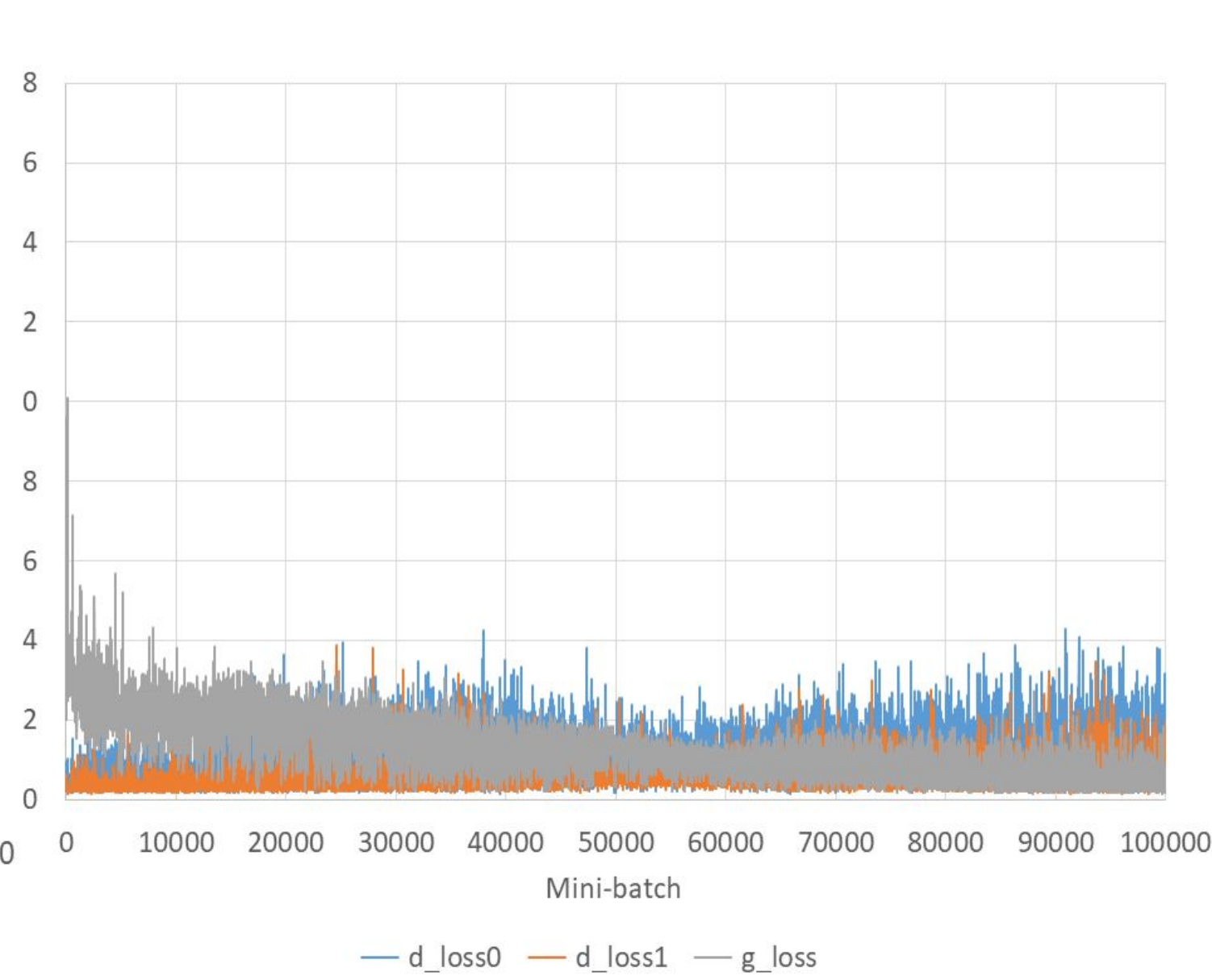
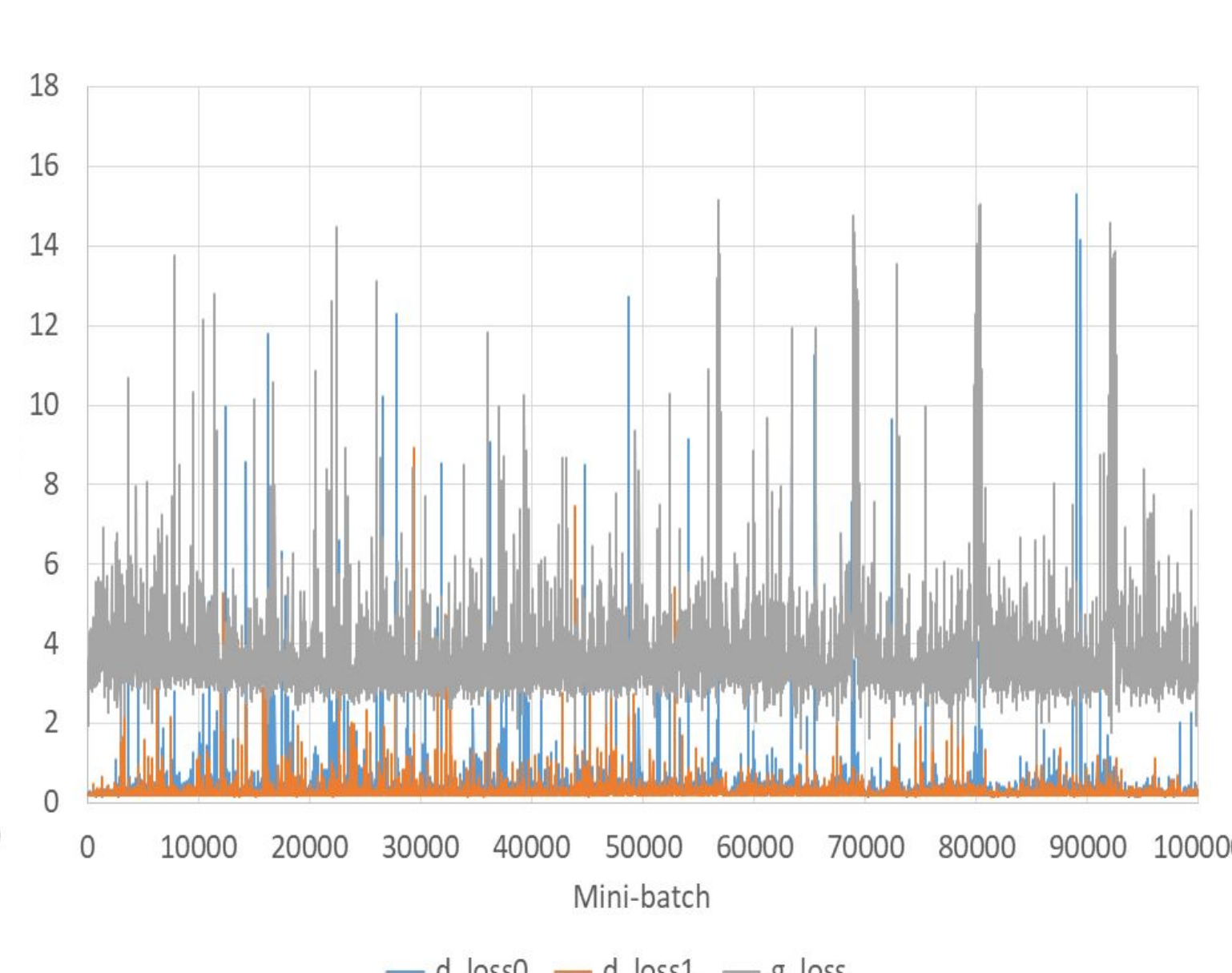
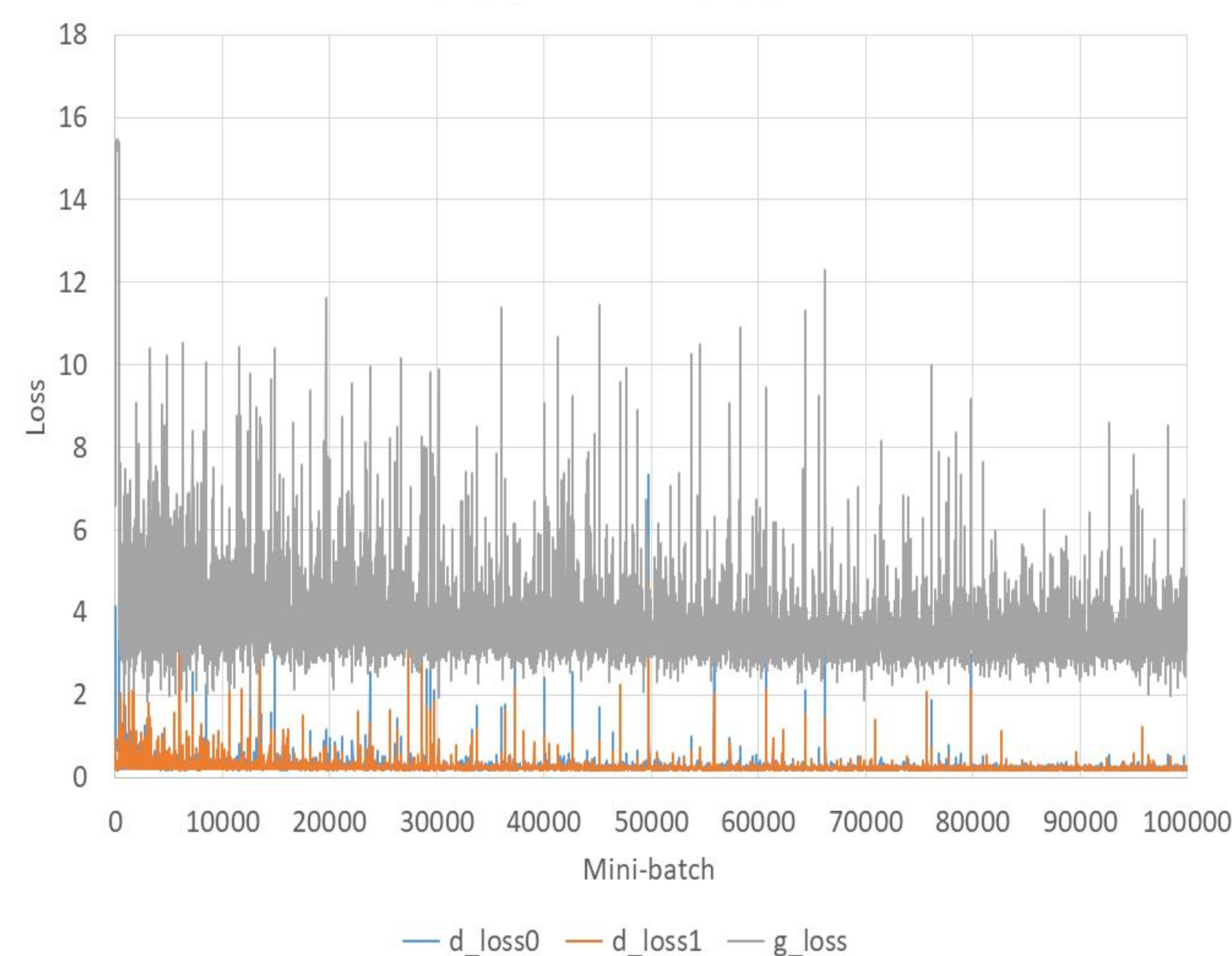


EXPERIMENTS - KERAS ANIME FACES

- Used different optimizers (SGD, Adam, Adamax)
- SGD for Discriminator, Adam for Generator
- Adamax for both
- Adjusted learning rates for both the discriminator and generator
- Adjusted batch sizes
- Added an additional layer
- Flipped labels for True and False



True vs False Labels Flipped



Adam, D LR=0.001, G LR=0.0001, $\beta_1=0.5$



Adamax, D LR=0.001, G LR=0.0001, $\beta_1=0.5$



Batch Size 16



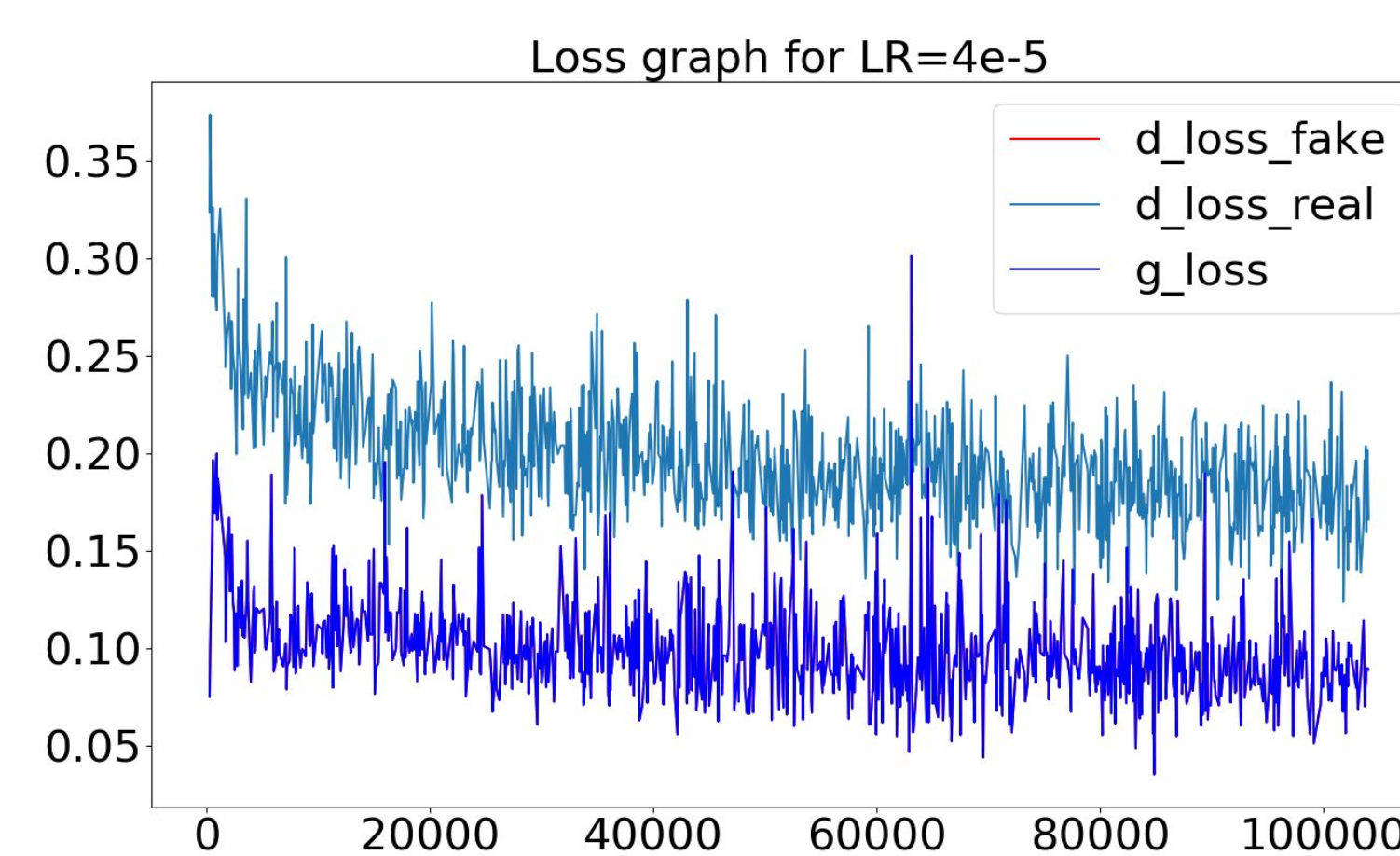
Batch Size 128

EXPERIMENTS - REAL FACE GANs

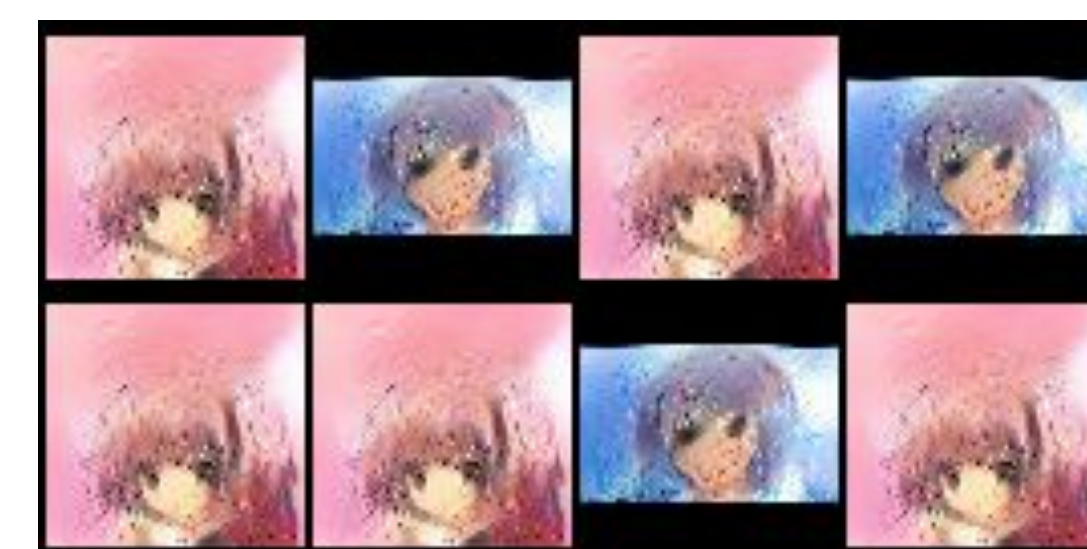
- **BEGAN-tensorflow**
 - BEGAN balances discriminator and generator
 - Blurs real image to keep discriminator balanced
 - Started overfitting
 - Lowered the learning rate (LR) and batch-size but still overfitting
 - Generator LR too low creates bad images
- **DCGAN Real Face**
 - Regularization caused generator to generate nearly identical images
 - Discriminator overpowers generator after 100 epoches
 - Limiting Discriminator training provided a better balance



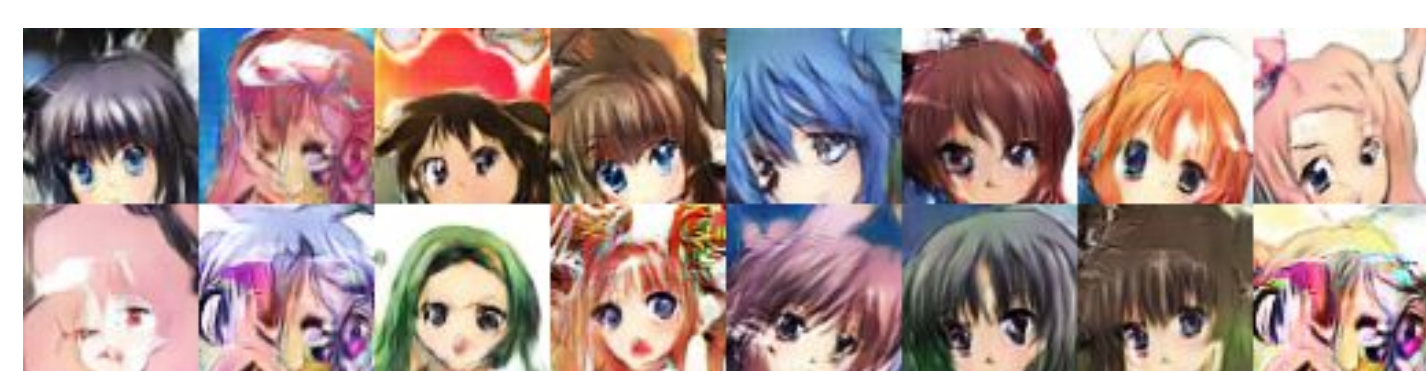
Generated Faces with LR=4e-5



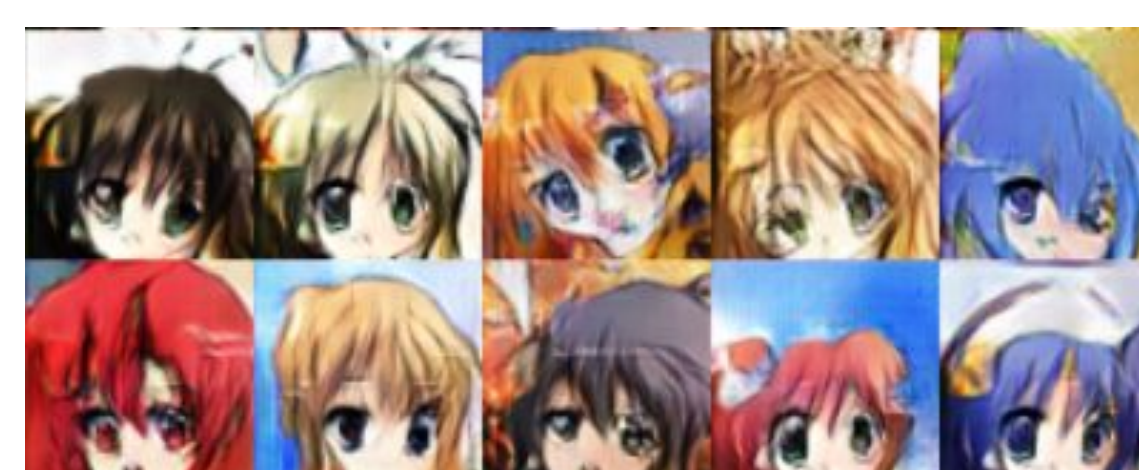
Real Anime Faces



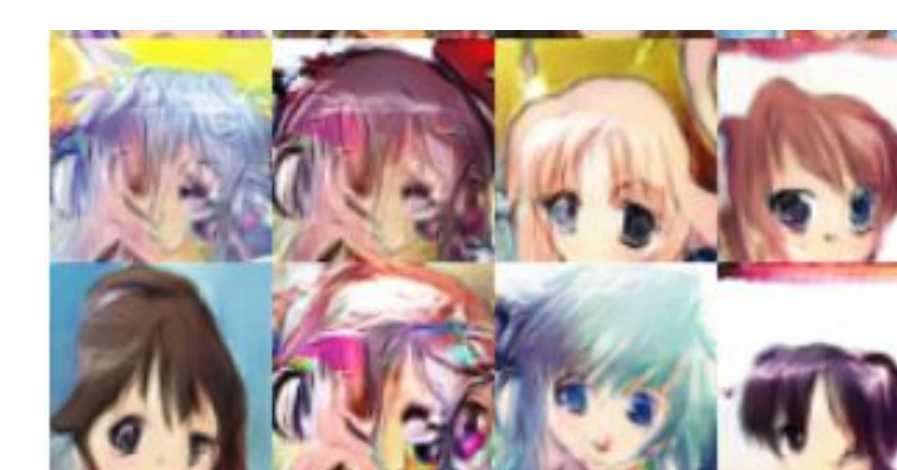
Initial Generated Faces with LR=8e-5



Generated Faces with LR=1e-4 and $\beta_1=0.5$



Generated Faces by limiting training on the more powerful network



Generated Faces without limiting training on the more powerful network