1. Show your model architecture and loss function.

* Encoder:

Conv2d(in\_channels=3, out\_channels=20, kernel\_size=3)

Relu()

Conv2d(in\_channels=20, out\_channels=2, kernel\_size=3)

Relu()

* Latent code:

加入zero mean & 0.01 variance的Gaussian noise (註: 只有在model inference階段才會加噪，於training階段沒有此指令)

* Decoder

ConvTranspose2d(in\_channels=2, out\_channels=20, kernel\_size=3)

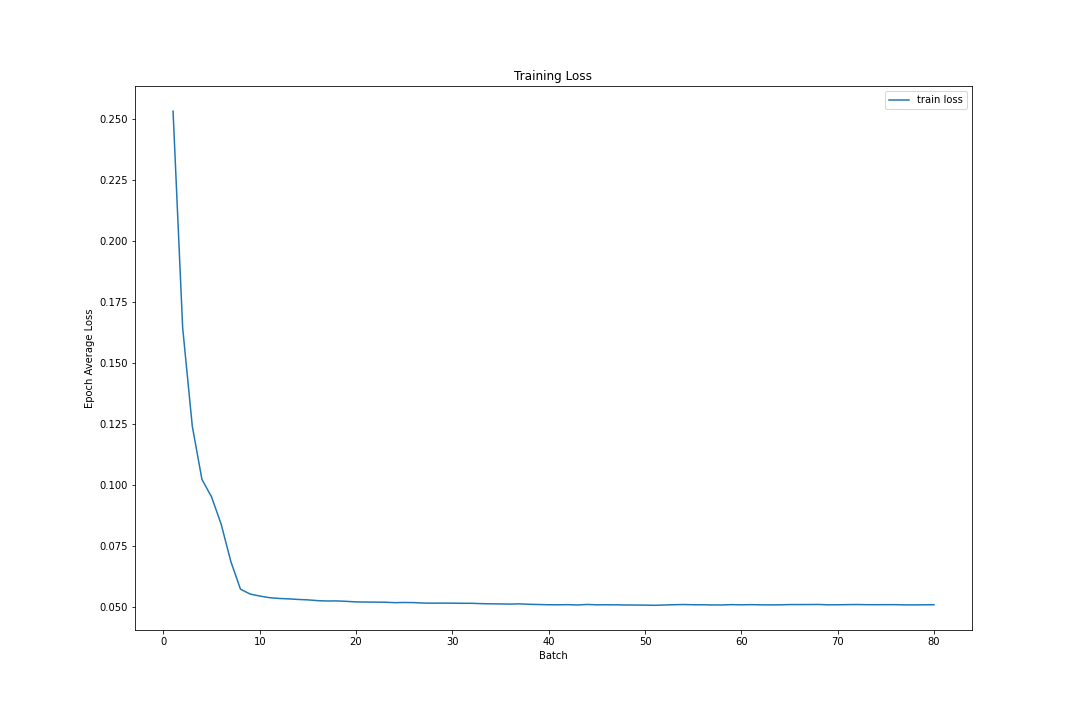
Relu()

ConvTranspose2d(in\_channels=20, out\_channels=3, kernel\_size=3)

Relu()

* Loss function: Mean square error

1. Plot training loss



1. Visualize 5 generated samples for each class

* Class 0 (Center)
  + Original



* + 5 generated samples

* Class 1 (Dount)
  + Original



* + 5 generated samples

* Class 2 (Edge-Loc)
  + Original



* + 5 generated samples

* Class 3 (Edge-Ring)
  + Original



* + 5 generated samples

* Class 4 (Loc)
  + Original



* + 5 generated samples

* Class 5 (Near-full)
  + Original



* + 5 generated samples

* Class 6 (Random)
  + Original



* + 5 generated samples

* Class 7 (Scratch)
  + Original



* + 5 generated samples

* Class 8 (None)
  + Original



* + 5 generated samples