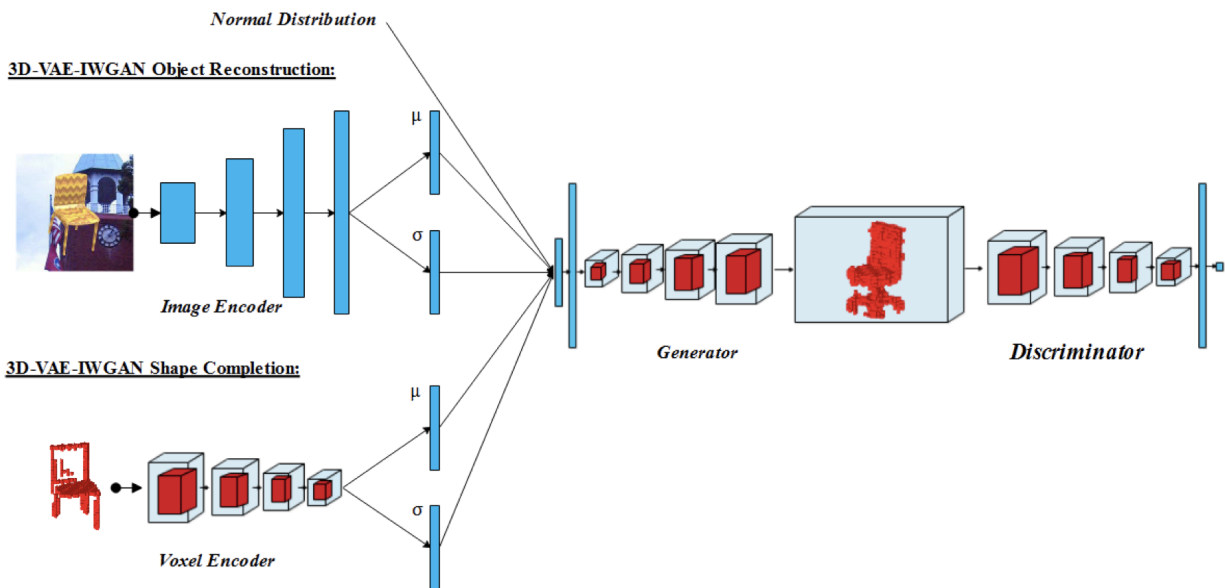# Improved Adversarial Systems for 3D Object Generation and Reconstruction

3D - VAE - IWGAN
Report

Dataset: ShapeNet, ModelNet, IKEA Dataset
Pytorch Code: https://github.com/EdwardSmith1884/3D-IWGAN
Paper: https://arxiv.org/pdf/1707.09557.pdf
Key points to note:

Encoder:

- The encoder observes samples from the target distribution and produces a vector of means and variances parameterizing a set of Gaussians, which are sampled to produce a latent vector.
- Basically, we try to sample in such a way that the latent vector is sampled from a normal distribution.

- Since in a normal GAN model, the input is from a Gaussian distribution we also make sure that the encoder also produces a latent vector similar to a Gaussian distribution so that the generator can learn easily
- The VAE's encoder converts an image into a 400-dimensional vector of means of variances, which are sampled using Gaussians to produce our latent vector.

Generator/Decoder

- The 400-dimensional vector is fed into the Generator which gives the desired 3D output.
- The generator is made to learn every 5 batches whereas the encoder/discriminator are made to learn every batch. This leads to a more stable convergence.
- This last point is key to the integration of the systems as if the encoder is not trained alongside the discriminator at every iteration the system will not converge. This makes sense as both networks should learn similar features about the objects being created at approximately the same rate.

Discriminator

- In the previous method, we used gradient descent and hence it has problems while convergence and it leads to unstable learning and may result in vanishing/exploding gradients. Hence to solve this problem Wasserstein distance was used.
- The main key point is that this method penalises deviation of the discriminator's gradients from 1, as the gradients of a differentiable function are at most 1 if and only if it is a 1-Lipschitz function.

- This forces the discriminator to lie within the set of 1-Lipschitz functions. This constraint is a key in ensuring constructed Wasserstein distance is always continuous and almost always differentiable.

Loss

- Discriminator's loss function:

$$\mathbb{E}_{\hat{x}\sim P_g}[D(\hat{x})] - \mathbb{E}_{x\sim P_r}[D(x)] + \lambda\mathbb{E}_{\hat{x}\sim P_x}[(||\nabla_{\hat{x}}D(\hat{x})||_2 - 1)^2]$$

- Encoder's loss function:

$$||\hat{x} - x||_2 + \mathrm{KL}[N(\mu, \Sigma)||N(0, I)]$$

- Generator's loss Function:

$$D(\hat{x}) + \delta||\hat{x} - x||_2$$

- Where x is the target sample, x^ is the generated sample (generated from an encoded image in the first equation, and a random latent vector in the second), μ and Σ are the means and variances produced by the encoder, and δ = 100