

Contrastive Convolutional Auto-Encoder(CCAE)

Aniket Nath, Diptarko Choudhury

School of Physical Sciences

National Institute of Science Education and Research, Bhubaneswar

Homi Bhaba National Institute

October 17, 2023

Introduction

We are interested in image generation, using AI techniques.

AI generating ...



Image generation has many application, in fields of data augmentation, super resolution etc.

Different approaches to the problem

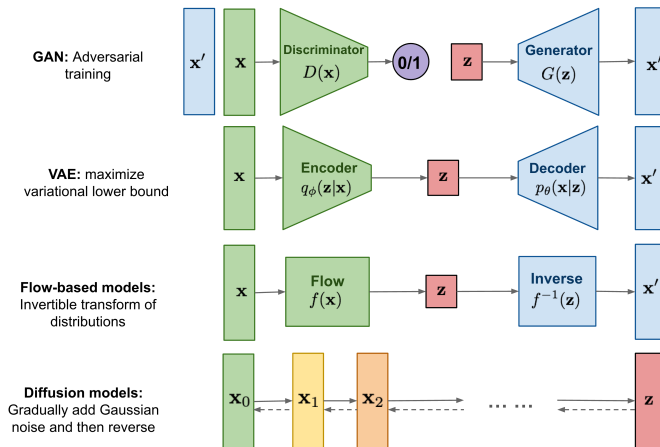


Figure: Different approaches to the problem

Survey of different models

- **Auto-Encoder:** Gives identity maps, Invertible, good reconstruction, not suitable for new image generation.
- **Variational Auto-Encoder:** Identity maps, Invertible, Bad reconstruction, suitable for new image generation.
- **Generative Adversarial Networks:** Good image generation, non-identity map, non-trivial inversion, suitable for generation of new samples, difficult to train.
- **Invertible Generative Adversarial Networks:** Good image generation, Identity map, difficult to train and invert.
- **Diffusion model:** Non-trivial inversion, no latent space in architecture.

VICReg Architecture

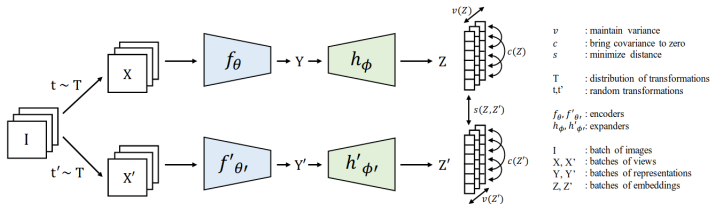


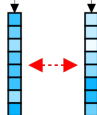
Figure: The VICReg architecture

$$\text{Encoder } f_{enc} : \mathbb{R}^{c \times h \times w} \rightarrow \mathbb{R}^{2048} \quad (1)$$

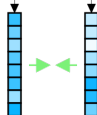
$$\text{Decoder } f_{dec} : \mathbb{R}^{2048} \rightarrow \mathbb{R}^{c \times h \times w} \quad (2)$$

$$\text{Projector } f_{proj} : \mathbb{R}^{2048} \rightarrow \mathbb{R}^{8192} \quad (3)$$

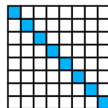
Mathematical description and VICReg Loss



Variance



Invariance



Covariance

$$Y = f_{enc}(X) \quad (4)$$

$$Z = f_{proj}(Y) \quad (5)$$

$$X' = f_{dec}(Y) \quad (6)$$

$$Y' = f_{enc}(X') \quad (7)$$

$$Z' = f_{proj}(Y') \quad (8)$$

Architecture I for Image generation

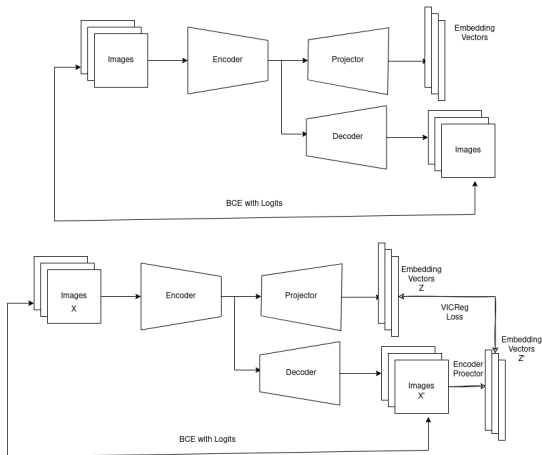


Figure: Architecture for image generation

Results



Figure: Input and Reconstructed image with Convolutional Autoencoders (above), Input and Reconstructed image using VICReg inversion(below)

Loss

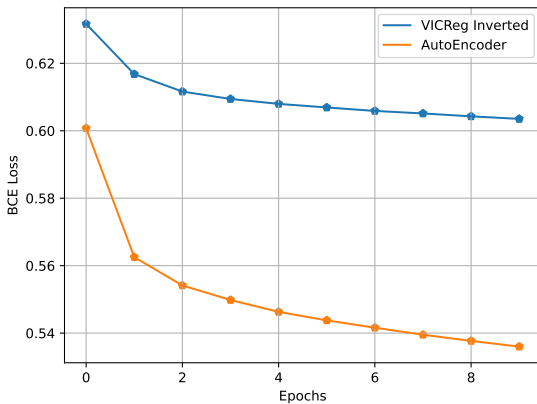


Figure: Comparing the losses for the different architectures