DEEP LEARNING MIDTERM REPORT

Variational autoencoder GAN for medical image generation.

Group members

Nguyễn Thái Bình - 22BI13059

Nguyễn Minh Đức - 22BI13092

Vũ Tuấn Hải - 22BI13149

Cấn Minh Hiển - 22BI13154

Nguyễn Quang Hưng - 22BI13184

Nguyễn Thế Khải - 22BI13201

Table of Contents

1. Introduction

Medical imaging provides essential insights into the body's internal structures but is challenged by limited high-quality data, privacy concerns, and the need for advanced diagnostic tools. VAE is effective at learning compact representations of complex data, while GANs excel at generating realistic synthetic images. Merging these methods can leverage their strengths to develop a powerful tool to enhance medical image generation and analysis.

In this project, we aim to apply VAE-GAN models to generate brain tumor MRI scans, providing an accurate diagnostic tool for identifying various brain conditions like cancer, cerebral infarction, encephalocele, and more.

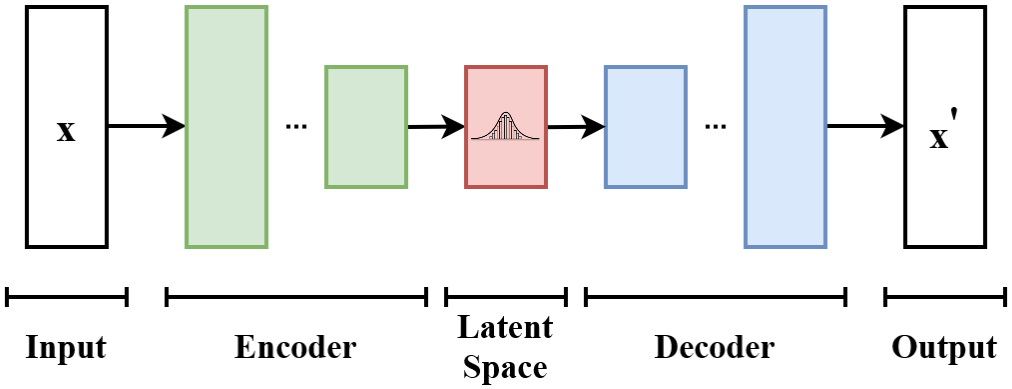
2. Deep Learning Model: VAE – GAN

2.1. Variational Autoencoders (VAEs)

Variational Autoencoders is an artificial neural network architecture designed to generate new data points similar to the input data by learning a probabilistic distribution over the data.

- The **Encoder** extracts latent variables of input data x and outputs them in the form of a vector representing latent space z.

- The **Latent space** is both the output layer of the encoder network and the input layer of the decoder network. It is fully compressed, lower-dimensional embedding of the input data.
- The **Decoder** use the data in latent space to reconstruct the original input by essentially reversing the encoder



Loss function:

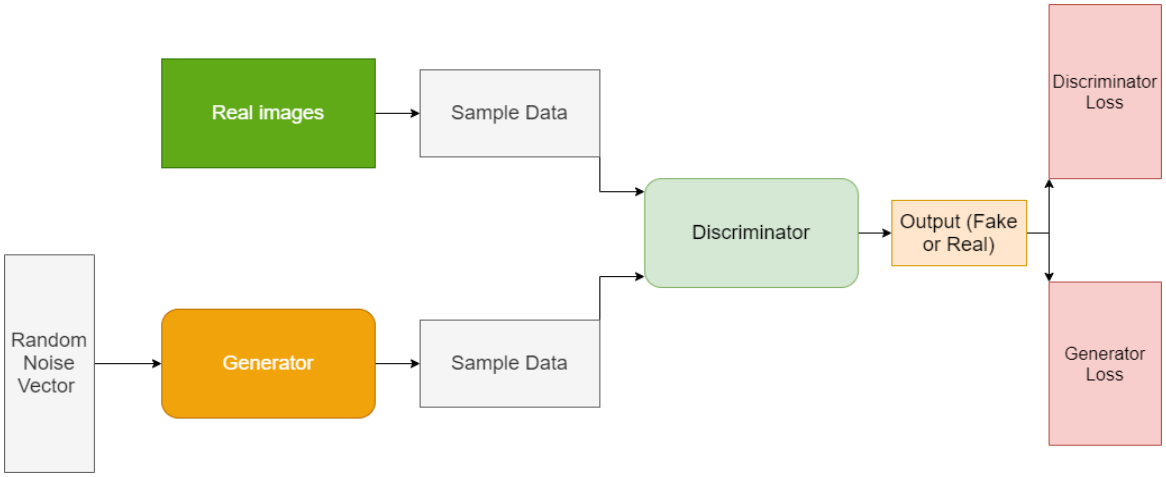
$$\mathcal{L}(\theta, \varphi; x, z) = \underbrace{\mathbb{E}_{q_\varphi(z|x)}[\log p_\theta(x|z)]}_{\textbf{Reconstruction loss}} - \underbrace{D_{KL}(q_\varphi(z|x) \,||\, p(z))}_{\textbf{Kullback-Leibler divergence}}$$

Reconstruction loss: is an expectation operator that measures how close the decoder output is to the original input.

Kullback-Leibler divergence: measures the difference between two probability distributions, forcing latent distribution to stay close to Normal (0,1).

2.2. Generative Adversarial Networks (GANs)

Generative Adversarial Networks are a deep neural network that can learn from training data and generate new data with the same characteristics. A generative adversarial network is made



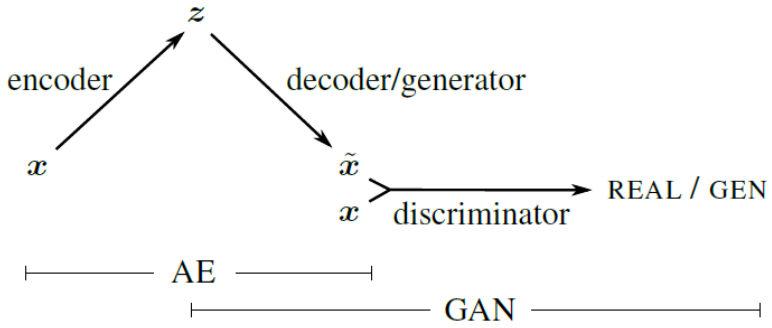
up of two neural networks, which are trained simultaneously, with the generator trying to fool the discriminator and the discriminator trying to classify real and fake samples accurately.

- **The Generator** takes random noise as input and produces data from it. Its goal is to generate data that is as real as possible.
- **The Discriminator** takes real data and the data generated by the Generator as input and attempts to distinguish between the two. It outputs the probability that the given data is real.

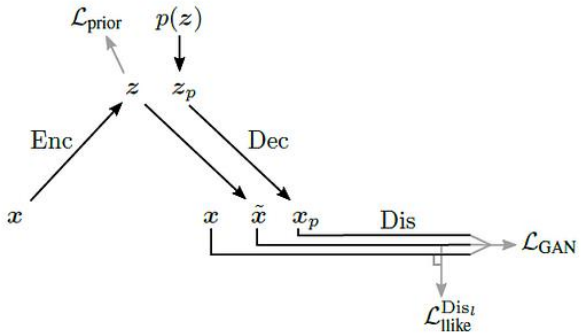
Loss function:

$$\min(G)\max(D)\,V(D,G) = E_{x\sim p_{data}(x)}[\log D(x)] + E_{x\sim p_z(z)}[\log(1\text{-}D(G(z))]$$

2.3. VAE – GAN Hybrid



A VAE is combined with a GAN by collapsing the decoder and the generator into one.



Algorithm 1 Training the VAE/GAN model

$\theta_{\text{Enc}}, \theta_{\text{Dec}}, \theta_{\text{Dis}} \leftarrow$ initialize network parameters
repeat
  $\boldsymbol{X} \leftarrow$ random mini-batch from dataset
  $\boldsymbol{Z} \leftarrow \text{Enc}(\boldsymbol{X})$
  $\mathcal{L}_{\text{prior}} \leftarrow D_{\text{KL}}(q(\boldsymbol{Z}|\boldsymbol{X})\|p(\boldsymbol{Z}))$
  $\tilde{\boldsymbol{X}} \leftarrow \text{Dec}(\boldsymbol{Z})$
  $\mathcal{L}_{\text{llike}}^{\text{Dis}_l} \leftarrow -\mathbb{E}_{q(\boldsymbol{Z}|\boldsymbol{X})}[p(\text{Dis}_l(\boldsymbol{X})|\boldsymbol{Z})]$
  $\boldsymbol{Z}_p \leftarrow$ samples from prior $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$
  $\boldsymbol{X}_p \leftarrow \text{Dec}(\boldsymbol{Z}_p)$
  $\mathcal{L}_{\text{GAN}} \leftarrow \log(\text{Dis}(\boldsymbol{X})) + \log(1 - \text{Dis}(\tilde{\boldsymbol{X}})) + \log(1 - \text{Dis}(\boldsymbol{X}_p))$
  // Update parameters according to gradients
  $\theta_{\text{Enc}} \overset{+}{\leftarrow} -\nabla_{\theta_{\text{Enc}}}(\mathcal{L}_{\text{prior}} + \mathcal{L}_{\text{llike}}^{\text{Dis}_l})$
  $\theta_{\text{Dec}} \overset{+}{\leftarrow} -\nabla_{\theta_{\text{Dec}}}(\gamma\mathcal{L}_{\text{llike}}^{\text{Dis}_l} - \mathcal{L}_{\text{GAN}})$
  $\theta_{\text{Dis}} \overset{+}{\leftarrow} -\nabla_{\theta_{\text{Dis}}}\mathcal{L}_{\text{GAN}}$
until deadline

VAE-GAN Training Procedures

Loss function:

$$\mathcal{L} = \mathcal{L}_{prior} + \mathcal{L}_{llike}^{Disl} + \mathcal{L}_{GAN}$$

With

$$\mathcal{L}_{prior} = D_{KL}(q(z|x) \,||\, p(z))$$

$$\mathcal{L}_{llike}^{Dis_l} = -\mathbb{E}_{q(z|x)}[\log p(Dis_l(x)|\, z)]$$

$$\mathcal{L}_{GAN} = \log\big(Dis(x)\big) + \log(1 - Dis(Gen(z)))$$

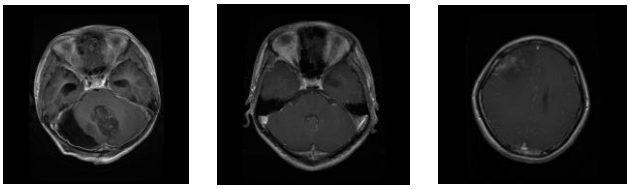
3. Experimentation

The model was implemented in Pytorch, using NVIDIA CUDA for acceleration and efficient training on high dimensional images.

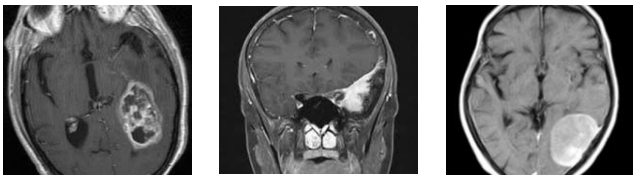
3.1. Dataset

This project uses a dataset containing various brain tumor MRI images. A brain tumor is a collection, or mass, of abnormal cells in the brain. When brain tumors grow, they can cause brain damage and even be life-threatening. The dataset includes over 5000 images categorized into the following classes:

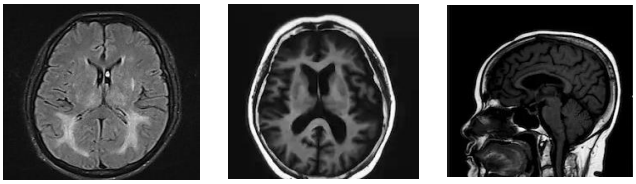
- *Glioma*: A glioma is a type of primary tumor that starts in the glial cells of the brain or spinal cord.



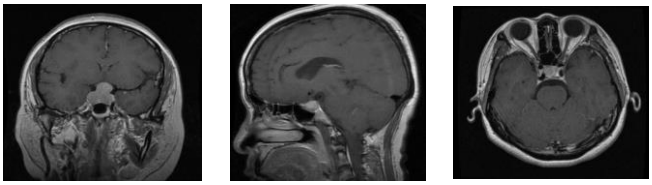
- *Meningioma*: Meningioma is typically a slow-growing tumor from the meninges, the membranous layers surrounding the brain and spinal cord.



- *Notumor*: No tumor is when the brain is in standard condition and no tumor appears.



- *Pituitary*: The pituitary gland is a small, pea-sized endocrine gland located at the base of the brain below the hypothalamus.



3.2. Model Components:

Encoder	Decoder	Discriminator
5x5 64 conv. ↓, BNorm, ReLu	8x8x256 fully-connected, BNorm,	5x5 32 conv. ↓, BNorm, ReLu
5x5 128 conv. ↓, BNorm, ReLu	ReLu	5x5 128 conv. ↓, BNorm, ReLu
5x5 256 conv. ↓, BNorm, ReLu	6x6 256 conv. ↑, BNorm, ReLu	5x5 256 conv. ↓, BNorm, ReLu
2048 fully-connected, BNorm, ReLu	6x6 128 conv. ↑, BNorm, ReLu	5x5 256 conv. ↓, BNorm, ReLu
	6x6 32 conv. ↑, BNorm, ReLu	512 fully-connected, BNorm, ReLu
	5x5 1 conv., tanh	1 fully-connected, sigmoid

Architectures for the three networks (Encoder, Decoder, Discriminator) that comprise VAE/GAN. ↓ and ↑ represent down- and upsampling respectively. BNorm denotes batch normalization.

3.3. Training Setup

- Latent space: The Encoder outputs a 128-dimensional latent vector.
- Optimizers: RMSProp was used with a learning rate of 3e-4 for the encoder and the decoder; and 3e-5 for the discriminator.
- Number of epochs: 25.
- Batch size: 64.
- The image is resized to 64x64 pixels, converted to Pytorch tensor then normalize the pixel values to a range [-1; 1].

3.4. Devices

- The model was trained on an NVIDIA RTX 2060 and completed after 2 hours and 18 minutes. It was then saved to "*vae_gan_model.pth*", allowing it to run on any device without retraining. This helps low-end devices handle the model without any issues.

4. Result and Analysis

4.1. Result

- The generated images have a general structural consistency, meaning the VAE-GAN model has learned key spatial features from the dataset.
- There are variety in the intensity and texture across the images, indicating that the model has captured the variability in the training data.

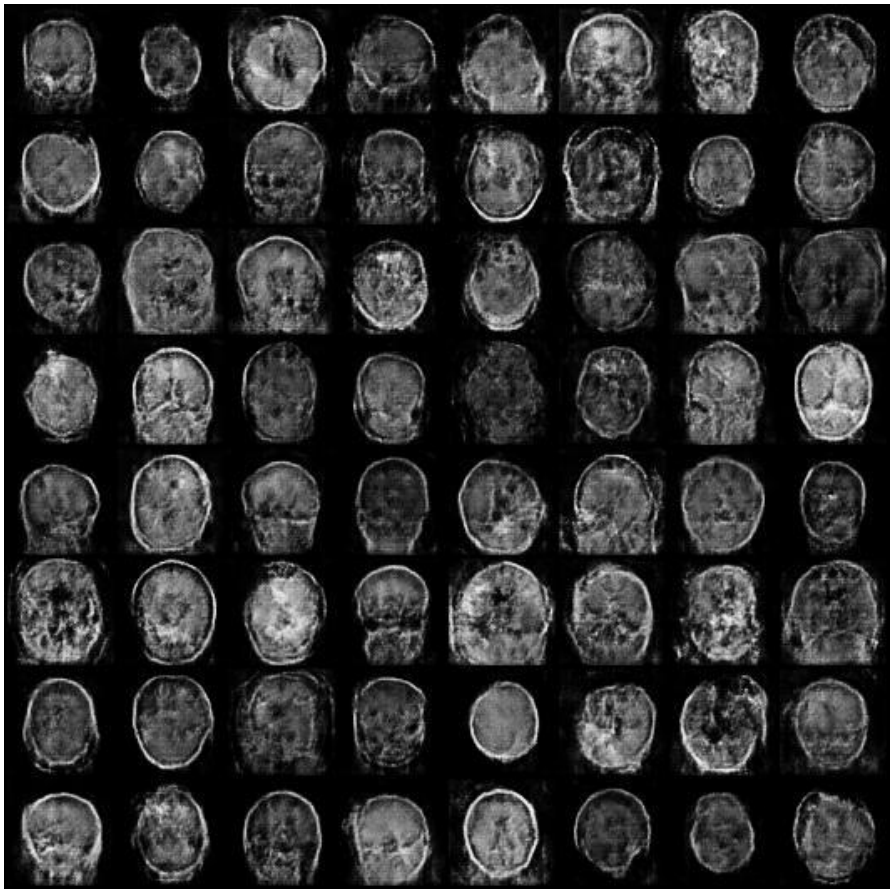


Image generated by VAE – GAN model

4.2. Analysis

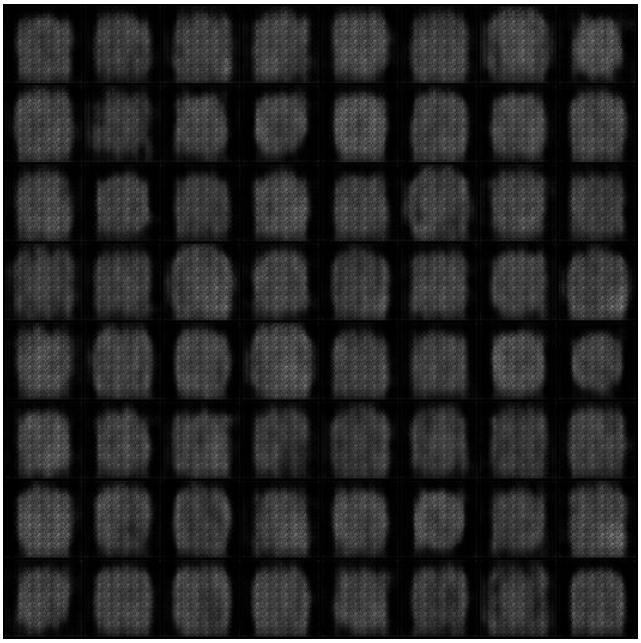


Image1: after 1 epoch

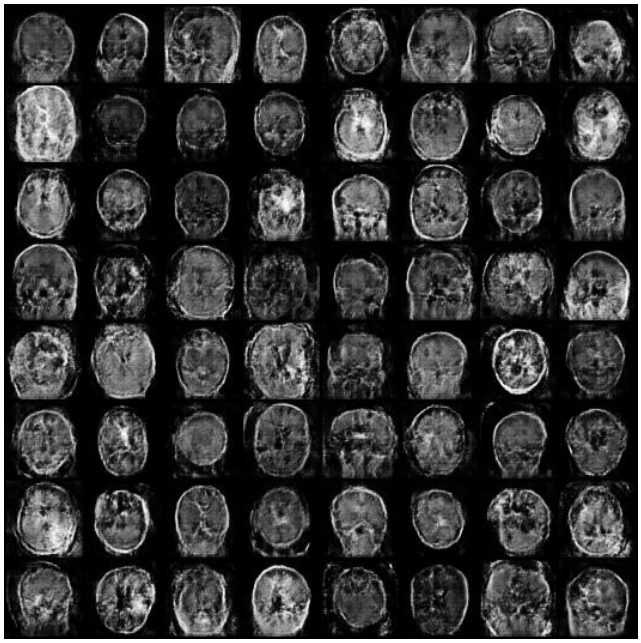


Image2: after 25 epochs

- Image 1: The generated images are highly blurry and lack distinct features. This mean model has just started training and has not learned any meaningful patterns from the data.

- Image 2: The generated images show significant improvement. The images now have clearer structures, with recognizable patterns and features that resemble real images. While there is still some noise and the images are not perfect, they are far more detailed than those generated after 1 epoch.

The VAE-GAN model shows noticeable progression in image generation from epoch 1 to epoch 25. Initially, the generated images are very blurry with no clear features, which means the model has not yet learned the patterns of the data. By epoch 25, the generated images show significant improvement, capturing the overall structure and texture of the brain MRI scans. However, noise is still present, so further adjustments can improve the image details.

5. Conclusion

Integrating Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs) in brain tumor MRI generation marks a breakthrough in medical imaging AI. By combining the structured latent space of VAEs with the realistic image generation capabilities of GANs, this model supports large-scale research and data sharing while preserving patient privacy and fostering medical innovation.

In summary, we have successfully demonstrated unsupervised learning using encoder-decoder models and a similarity measure. Our results indicate that the visual fidelity of our method is very promising for medical researchers. However, our model requires further fine-tuning to enhance diagnostic accuracy, reduce risk, and improve patient outcomes in brain tumor detection and treatment.

6. References

- A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," *arXiv*, 2015. [Online]. Available: https://arxiv.org/pdf/1512.09300
- S. H. Tsang, "Review: VAE-GAN - Autoencoding beyond pixels using a learned similarity metric," *Medium*, Oct. 10, 2019. [Online]. Available: https://sh-tsang.medium.com/review-vae-gan-autoencoding-beyond-pixels-using-a-learned-similarity-metric-dc0f8cb74435
- D. Bergmann and C. Stryker, "Variational autoencoder," *IBM*, June 12, 2024. [Online]. Available: https://www.ibm.com/think/topics/variational-autoencoder
- P. D. Khanh, "GAN: An overview," *phamdinhkhanh.github.io*, July 13, 2020. [Online]. Available: https://phamdinhkhanh.github.io/2020/07/13/GAN.html
- M. Del Pra, "Generative adversarial networks," *Medium*, Oct. 30, 2023. [Online]. Available: https://medium.com/@marcodelpra/generative-adversarial-networks-dba10e1b4424