1 Paper Introduction

This homework explores Latent Diffusion Models as presented in the paper High Resolution Image Synthesis with Latent Diffusion. Stable diffusion is a text conditional latent diffusion model, created by researchers and engineers from CompVis, Stability AI, and LAION. Diffusion models are incredibly expensive to the train so with the use of latent space, we can greatly reduce the computational resource requirements for training, and also result very competitive performance on tasks such as inpainting, text-to-image synthesis, super-resolution and more. We encourage you to read over the paper on your own as it is quite fascinating!

1.1 Auto-encoders and Downsampling

Latent diffusion models use pre-trained auto-encoders to produce a lower dimensional space representation of the input. Auto-encoders downsample the input images by a factor of $f = 2^m, m \epsilon \mathbb{R}$. The paper experiments with 6 different sampling factors $f = 1, 2, 4, 8, 16, 32$ abbrieviated as LDM-f. LDM-1 represents pixel-based diffusion models with no downsampling. The paper tested the performance on these 6 models and measured performance with two different scores: FID (Frechet Inception Distance) which compares the distributions of images generated with the model and real images and IS (Inception Score) which quantifies the quality of images generated by the model.

(a) Given two graphs that show the FID scores and IS scores against training progress, we can see that three of the LDM-f models perform well and three have weaker performance. Identify the top three performing LDM-f models and the bottom three performing LDM-f models. Provide a brief explanation for your answer.

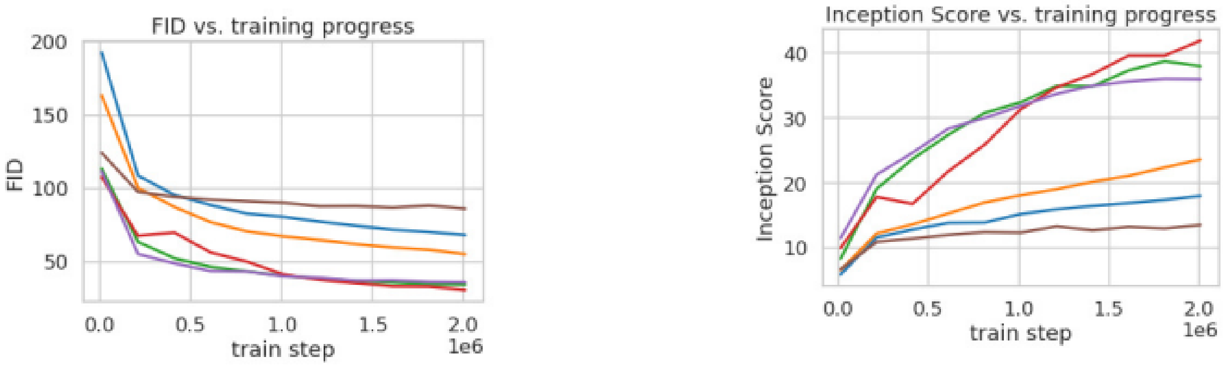

Abbildung 1: IS Scores vs Training Progress

Solution: For FID, the lower the FID score, the greater the similarity between the distributions of the generated images and real images. For the inception score, the higher the score, the better the quality of the model. The top three performing models are LDM-4, LDM-8, LDM-16. The worst of the three models are LDM-1, LDM-2, and LDM-32. With LDM-1 and LDM-2, the performance is too slow and we see the same issues that occur with the large training time and computation requirements of regular diffusion models. LDM-32 results in information loss due to too much compression. With a downsampling factor of $f = [4, 16]$ we can reap the benefits of faster performance without having losing too much information about the image in the process.

1.2 Conditional LDM Loss

We can think of diffusion models as a sequence of denoising autoencoders $\epsilon_\theta(x_t, t)$; $t = 1...T$ where x_t is a noisy version of the input and t is sampled uniformly. We get the following loss:

$$L_{LDM} = \mathbb{E}_{\varepsilon(x),\varepsilon\sim\mathcal{N}(0,1),t}[\|\varepsilon - \varepsilon_\theta(x_t, t)\|_2^2]$$

For a conditional LDM, we use a conditional denoising autoencoder which takes in the latent representation, z_t and a domain specific encoder τ_θ that projects a text input y into an intermediate representation.

(b) Write the loss function for a conditional LDM.

Solution: Instead of taking in the noisy input x_t, we take in the latent representation and the intermediate representation of a text input y after being passed through the domain specific encoder, τ_θ. The output of the conditional denoising autoencoder can be written as, $\varepsilon_\theta(z_t, t, \tau_\theta(y))$. Thus the loss can be written as

$$L_{LDM} = \mathbb{E}_{\varepsilon(x),y,\varepsilon\sim\mathcal{N}(0,1)}[\|\varepsilon - \varepsilon_\theta(z_t, t, \tau_\theta(y))\|_2^2]$$

2 Image Synthesis with Diffusion Model

Please follow the instructions in this notebook. You will implement a text conditioned denoising unet and the sampling algorithm. Then you'll see the diffusion process on a single image and try to train your own diffusion mode. Once you finished with the notebook,

- download submission.zip and submit it to Gradescope.
- Answer the following questions in your submission of the written assignment:

(a) Why might a U-Net by a good choice for the model backbone? List two reasons.

Solution: (This is an open question, if the explanation of the answer is reasonable, then it can be considered correct)

- The model must output the amount of noise in the input, which is of the same shape as the input, meaning both the input and output are of identical shape. This is exactly what a U-Net does.
- UNet has an inductive bias geared towards image-like data. This is due to using convolutions and the downsampling followed by upsampling with skip connections between them which gives it the ability to pick out hierarchical and spatial structures in data.

(b) Screenshot your visualization for get noisy image and include it in your submission of the written assignment. And describe the picture you observed briefly. What kind of process is this?

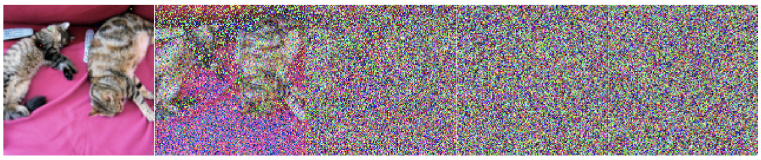

Abbildung 2: Visualization for Sampling

Solution: Shown as Abbildung 2. This should be a process of gradually adding noise to the original picture. Corresponding to the q sample in diffusion model algorithm.

(c) Screenshot one of your visualizations about the sampled images and include it in your submission of the written assignment. Then answer the following question. How does the model perform and does it meet your expectations? If not, what do you think are the directions for improvement?

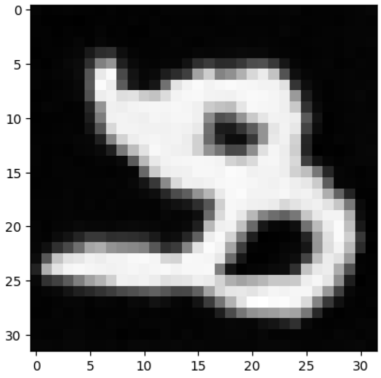

Abbildung 3: Visualization for Model Performance

Solution: The performance of the model was not so great and did not meet expectations. As can be seen from the visualization picture, it only learns the style of the dataset, but does not learn the real semantic information. The main reason is that the datasets and models we use are too small. At the same time, MNIST is not a standard image-text pair dataset, so our text definition is not very accurate. In addition, clip does not perform particularly well with handwritten digits. In order to improve the performance of the model, we also need to consider using a larger and more professional image-text pair dataset while increasing the scale of the model. This further underscores the current trend towards larger models.

3 Understanding the VAE Loss

Latent diffusion models use an autoencoder model similar to a VAE to encode images from pixel space to some compressed latent space, perform the diffusion process in this latent space, and finally decode the denoised latent back into pixel space.

To have our model learn the correct latent space, we would like the estimated posterior by the encoder $q_\phi(z|x_i)$ to be very close to the real posterior $p_\theta(z|x_i)$. We can quantify this using the Kullback-Leibler (KL) divergence to get a measure of the difference between the two distributions, defined as

$$D_{KL}(P||Q) = \mathbb{E}_{x\sim P(x)}[log\frac{P(x)}{Q(x)}]$$

However, $p_\theta(z|x_i)$ is intractable in practice. Another goal of VAE-like models is to maximize the probability of generating real samples, or $p_\theta(x)$. During training, rather than directly maximizing this quantity, we maximize a lower-bound called the evidence lower bound (ELBO):

$$log(p_\theta(x_i)) \geq \mathcal{L}(x_i, \theta, \phi) = \mathbb{E}_{z\sim q_\phi(z|x_i)}[log(p_\theta(x_i|z)] - D_{KL}[q_\phi(z|x_i)||p_\theta(z)]$$

At a high-level, this loss can be seen as doing two things:

- **Reconstruction loss**: The first term corresponds to the likelihood of generating images from the true data distribution given a sampled latent
- **Matching Prior**: The second term acts as a regularizer to minimize the difference between the estimated posterior and the prior (generally a standard normal).

Note: In the latent diffusion paper, the autoencoder loss is actually comprised of 3 terms: a patch-based discriminator loss for reconstruction, the standard KL-divergence term between $q_\phi(z|x)$ and a prior $p(z)$, and a regularizing term L_{reg} for the latent z to be zero centered and with small variance. For simplicity, however, we will just be analyzing the traditional ELBO loss here.

(a) **Show** that maximizing the ELBO loss also minimizes $D_{KL}(q_\phi(z|x_i)||p_\theta(z|x_i))$, the KL-divergence of the estimated posterior and the real posterior.

Hint: Start with the KL-divergence formula and manipulate it to get ELBO loss plus some other term.
Solution:

$$\begin{aligned}
D_{KL}(q_\phi(z|x_i)||p_\theta(z|x_i)) &= \mathbb{E}_{z\sim q_\phi(z|x_i)}[log\frac{q_\phi(z|x_i)}{p_\theta(z|x_i)}] && (1)\\
&= \mathbb{E}_{z\sim q_\phi(z|x_i)}[log\frac{q_\phi(z|x_i)p_\theta(x_i)}{p_\theta(z, x_i)}] && (2)\\
&= \mathbb{E}_{z\sim q_\phi(z|x_i)}[log\frac{q_\phi(z|x_i)p_\theta(x_i)}{p_\theta(x_i|z)p_\theta(z)}] && (3)\\
&= -\mathbb{E}_{z\sim q_\phi(z|x_i)}[log\, p_\theta(x_i|z)] + \mathbb{E}_{z\sim q_\phi(z|x_i)}[log\frac{q_\phi(z|x_i)}{p_\theta(z)}] + \mathbb{E}_{z\sim q_\phi(z|x_i)}[log\, p_\theta(x_i)] \\
& && (4)\\
&= -\mathbb{E}_{z\sim q_\phi(z|x_i)}[log\, p_\theta(x_i|z)] + D_{KL}[q_\phi(z|x_i)||p_\theta(z)] + log\, p_\theta(x_i) && (5)\\
&= -\mathcal{L}(x_i, \theta, \phi) + log\, p_\theta(x_i) && (6)\\
& && (7)
\end{aligned}$$

This gives $log\, p_\theta(x_i) = \mathcal{L}(x_i, \theta, \phi) + D_{KL}(q_\phi(z|x_i)||p_\theta(z|x_i))$. Since $log\, p_\theta(x_i)$ is independent of q_ϕ, and thus fixed, maximizing the ELBO loss minimizes the KL term. This tells us that maximizing the ELBO loss does exactly what we want: maximize the likelihood of generating real data and minimize the difference between the estimated and real posterior distributions of the latent space.