Comparing Diffusion Methods for Image Generation

Neha Vinjapuri (nehavin@stanford.edu)
Stanford University

Motivation

Diffusion models have emerged as powerful generative models capable of producing high-quality images. These models work by iteratively denoising a randomly sampled noise image to generate realistic outputs. The ability to leverage diffusion models extends beyond generation and into solving inverse problems such as image inpainting and deconvolution. This poster explores how diffusion models can be used in these contexts and evaluates different methods for solving these tasks.

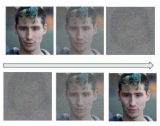
Methods

Methods:

1) Baseline DDPM Sampling: Vanilla iterative denoising.
2) SDEdit: Image modification via noise perturbation.
3) ScoreALD: Incorporates an annealing strategy for more stable reconstructions.
4) DPS: Fine-tuned posterior sampling method for improved consistency.

Dataset: Flickr-Faces-HQ Dataset (FFHQ)

Baseline DDPM:



Noising: Iteratively add noise for T steps

Denoising: Remove noise for T steps

SDEdit:

ScoreALD:

$$x_T \sim \mathcal{N}(0, \mathbf{I})$$
$$\textbf{for } t = T, \ldots, 1 \textbf{ do}$$
$$\mathbf{z} \sim \mathcal{N}(0, \mathbf{I}) \text{ if } t > 1, \text{else } \mathbf{z} = 0$$
$$\hat{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t + (1-\bar{\alpha}_t)s_\theta(x_t, t))$$
$$x_{t-1} = \frac{\sqrt{\bar{\alpha}_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\bar{\alpha}_t)}{1-\bar{\alpha}_t}\hat{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$$
$$x_{t-1} = x_{t-1} - \frac{\rho_t}{\|y - A\hat{x}_0\|_2}\nabla_{x_t}\|y - A(\hat{x}_0)\|_2^2$$
$$\textbf{end for}$$
$$\textbf{return } x_0$$

Naive assumption:

$$p(y|x_t) \cong p(y|x_0)$$

Diffusion Posterior Sampling:

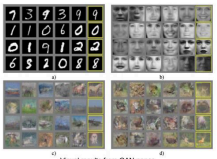
$$x_T \sim \mathcal{N}(0, \mathbf{I})$$
$$\textbf{for } t = T, \ldots, 1 \textbf{ do}$$
$$\mathbf{z} \sim \mathcal{N}(0, \mathbf{I}) \text{ if } t > 1, \text{else } \mathbf{z} = 0$$
$$\hat{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t + (1-\bar{\alpha}_t)s_\theta(x_t, t))$$
$$x_{t-1} = \frac{\sqrt{\bar{\alpha}_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\bar{\alpha}_t)}{1-\bar{\alpha}_t}\hat{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$$
$$x_{t-1} = x_{t-1} - \frac{\rho_t}{\|y\|^2}\nabla_{x_t}\|y - A(\hat{x}_0)\|_2^2$$
$$\textbf{end for}$$
$$\textbf{return } x_0$$

Improved assumption:

$$p(y|x_t) = \int p(y|x_0)p(x_0|x_t)dx_0$$

Related Work

1) **Generative Adversarial Networks**
   - **Approach**: Generator and Discriminator
   - **Limitations**: specific and hard to generalize for other tasks
2) **Variational Autoencoders**
   - **Approach**: Learn latent space, interpolate
   - **Limitations**: Less flexible, blurry images
3) **Autoregressive Models/PixelCNN**
   - **Approach:** generate each pixel w/ conditional probabilities
   - **Limitations:** slow sampling speed due to sequential generation



Visual results from GAN paper.

Experimental Results

Baseline DDPM:

PNSR: 28.595932896574296, LPIPS: 0.18288618001747131

PNSR: 32.458862342466524, LPIPS: 0.07835651934146881

Smoothing, loss of fine detail, great for retaining main components

Unconditional Generation:

Contains all components, uncanny features in the background and in details of face

SDEdit:

PSNR: 19.748328788115117, LPIPS: 0.26346722224521637

PSNR: 20.366582139797345, LPIPS: 0.2103722244501114

Other features regenerated, true to overall shape

ScoreALD:

PSNR: 27.176294919869548, LPIPS: 0.05317398697137833

Annealing Factor: 15-20

ScoreALD (Cont.):

PSNR: 29.263467222452163?, LPIPS: 0.21039869701114

Annealing Factor: 10-15

Removes blurry/blocked portions, sometimes unmatching features

DPS:

References

[1] Chung, H., et al. (2023). "Diffusion Posterior Sampling for General Noisy Inverse Problems."
[2] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). "Generative Adversarial Networks".
[3] Ho, J., Jain, A., Abbeel, P. (2020). "Denoising Diffusion Probabilistic Models."
[4] Jalal, A., et al. (2021). "Robust Blind Inverse Problems in Imaging with Score-based Generative Models."
[5] Kingma, D. P., & Welling, M. (2013). "Auto-Encoding Variational Bayes"
[6] Meng, C., et al. (2022). "SDEdit: Image Synthesis and Editing with Stochastic Differential Equations."
[7] van den Oord, A., Kalchbrenner, N., Vinyals, O., Espeholt, L., Graves, A., & Kavukcuoglu, K. (2016). "Conditional Image Generation with PixelCNN Decoders".