UDPM: Upsampling Diffusion Probabilistic Models

Shady Abu-Hussein and Raja Giryes

TEL AVIV UNIVERSITY
Pursuing the Unknown

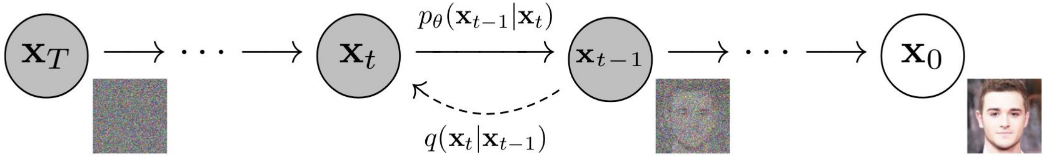
NEURAL INFORMATION PROCESSING SYSTEMS

Introduction

- Given a dataset of samples taken from an unknown distribution q(x)

- The target is to learn the data distribution q(x)
- Generate different samples from the same distribution q(x).

Denoising Diffusion Probabilistic Models



- Construct a Markovian diffusion process

Forward Process:

$$q(x_t|x_{t-1}) := \mathcal{N}(\alpha_t x_{t-1}, \sigma_t^2 I)$$

Backward Process:

$$p(x_{t-1}|x_t) := \mathcal{N}(\mu(x_t;t), \Sigma_t)$$

- The goal is to learn the distribution of the backward process

$$p_\theta(x_{0:T}) := p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)$$

- Although DDPM have shown great generation results, they have two major issues
  - → Long inference runtime (1000 denoising steps).
  - → Large and uninterpretable latent space.

UDPM: Upsampling Diffusion Probabilistic Models

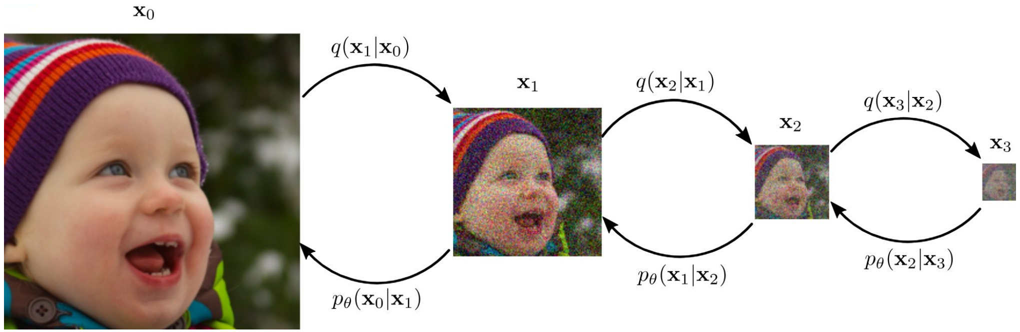
- Construct a diffusion process

Forward Process:

$$q(\mathbf{x}_l|\mathbf{x}_{l-1}) := \mathcal{N}(\alpha_l \mathcal{H}\mathbf{x}_{l-1}, \sigma_l^2 \mathbf{I}),$$

Backward Process:

$$p(\mathbf{x}_{l-1}|\mathbf{x}_l) := \mathcal{N}(\mu(\mathbf{x}_l; l), \Sigma_l),$$



- Training objective

$$D_{\mathrm{KL}}(q(\mathbf{x}_{1:L}|\mathbf{x}_0)||p_\theta(\mathbf{x}_{1:L}|\mathbf{x}_0)) := \mathbf{E}_q\left[\log \frac{q(\mathbf{x}_{1:L}|\mathbf{x}_0)}{p_\theta(\mathbf{x}_{1:L}|\mathbf{x}_0)}\right] = \log p_\theta(\mathbf{x}_0) \underbrace{-\mathbf{E}_q\left[\frac{p_\theta(\mathbf{x}_{0:L})}{q(\mathbf{x}_{1:L}|\mathbf{x}_0)}\right]}_{\text{ELBO}} \longrightarrow \ell_{\text{simple}}^{(l)} = \|f_\theta(\mathbf{x}_l) - \mathcal{H}^{l-1}\mathbf{x}_0\|_2^2$$

- In practice

$$\ell = \lambda_{\text{fid}}^{(l)} \ell_{\text{simple}} + \lambda_{\text{per}}^{(l)} \ell_{\text{per}} + \lambda_{\text{adv}}^{(l)} \ell_{\text{adv}}$$

UDPM: Upsampling Diffusion Probabilistic Models

- Similar to DDPM, we would like to choose H such that

$$q(\mathbf{x}_l|\mathbf{x}_0) = \mathcal{N}(\bar{\alpha}_l \mathcal{H}^l \mathbf{x}_0, \tilde{\sigma}_l^2 \mathbf{I}) \text{ where } \bar{\alpha}_l = \prod_{k=0}^{l} \alpha_k, \text{ and } \tilde{\sigma}_l = \bar{\alpha}_l^2 \sum_{k=1}^{l} \frac{\sigma_k^2}{\bar{\alpha}_l^2}$$

- Satisfied when the following *Lemma* holds

Lemma 1. *Let* $\mathbf{e} \overset{iid}{\sim} \mathcal{N}(0, \mathbf{I}) \in \mathbb{R}^N$ *and* $\mathcal{H} = \mathcal{S}_\gamma \mathcal{W}$*, where* $\mathcal{S}_\gamma$ *is a subsampling operator with stride* γ *and* $\mathcal{W}$ *is a blur operator with blur kernel* $\mathbf{w}$*. Then, if the support of* $\mathbf{w}$ *is at most* γ*, we have* $\mathcal{H}\mathbf{e} \overset{iid}{\sim} \mathcal{N}(0, \|\mathbf{w}\|_2^2 \mathbf{I})$.

- Using Bayes

$$q(\mathbf{x}_{l-1}|\mathbf{x}_l, \mathbf{x}_0) = \mathcal{N}(\mu(\mathbf{x}_l, \mathbf{x}_0, l), \Sigma_l),$$

where

$$\Sigma_l = \left(\frac{\alpha_l^2}{\sigma_l^2} \mathcal{H}^T \mathcal{H} + \frac{1}{\tilde{\sigma}_{l-1}} \mathbf{I} \right)^{-1},$$

and

$$\mu(\mathbf{x}_l, \mathbf{x}_0, l) = \Sigma_l \left(\frac{\alpha_l}{\sigma_l^2} \mathcal{H}^T \mathbf{x}_l + \frac{\bar{\alpha}_{l-1}}{\tilde{\sigma}_{l-1}^2} \mathcal{H}^{l-1} \mathbf{x}_0 \right).$$

UDPM - Overview

Algorithm 1 UDPM training algorithm

Require: $f_\theta(\cdot), L, q(\mathbf{x}), D_\phi(\cdot)$

while Not converged **do**

  $\mathbf{x}_0 \sim q(\mathbf{x})$

  $l \in \{1, 2, \dots, L\}$

  $\mathbf{e} \sim \mathcal{N}(0, I)$

  $\mathbf{x}_l = \bar{\alpha}_l \mathcal{H}^l \mathbf{x}_0 + \tilde{\sigma}_l \mathbf{e}$

  $\ell = \lambda_{\text{fid}}^{(l)} \ell_{\text{simple}} + \lambda_{\text{per}}^{(l)} \ell_{\text{per}} + \lambda_{\text{adv}}^{(l)} \ell_{\text{adv}}$

  ADAM step on θ

  Adversarial ADAM step on ϕ

end while

return $f_\theta(\cdot)$

Algorithm 2 UDPM sampling algorithm

Require: $f_\theta(\cdot), L$

$\mathbf{x}_L \sim \mathcal{N}(0, I)$

for all $l = L, \dots, 1$ **do**

  $\Sigma = \left(\frac{\alpha_l^2}{\sigma_l^2} \mathcal{H}^T \mathcal{H} + \frac{1}{\tilde{\sigma}_{l-1}^2} \mathbf{I} \right)^{-1}$

  $\mu_\theta = \Sigma \left[\frac{\alpha_l}{\sigma_l^2} \mathcal{H}^T \mathbf{x}_l + \frac{\bar{\alpha}_{l-1}}{\tilde{\sigma}_{l-1}^2} f_\theta^{(l)}(\mathbf{x}_l) \right]$

  $\mathbf{x}_{l-1} \sim \mathcal{N}(\mu_\theta, \Sigma)$

end for

return $\mathbf{x}_0$

UDPM - Results

Figure 1: Generated 64 × 64 images of AFHQv2 with **FID=7.10142**, produced using unconditional UDPM with only 3 steps, which are equivalent to 0.3 of a single typical 64 × 64 diffusion step.

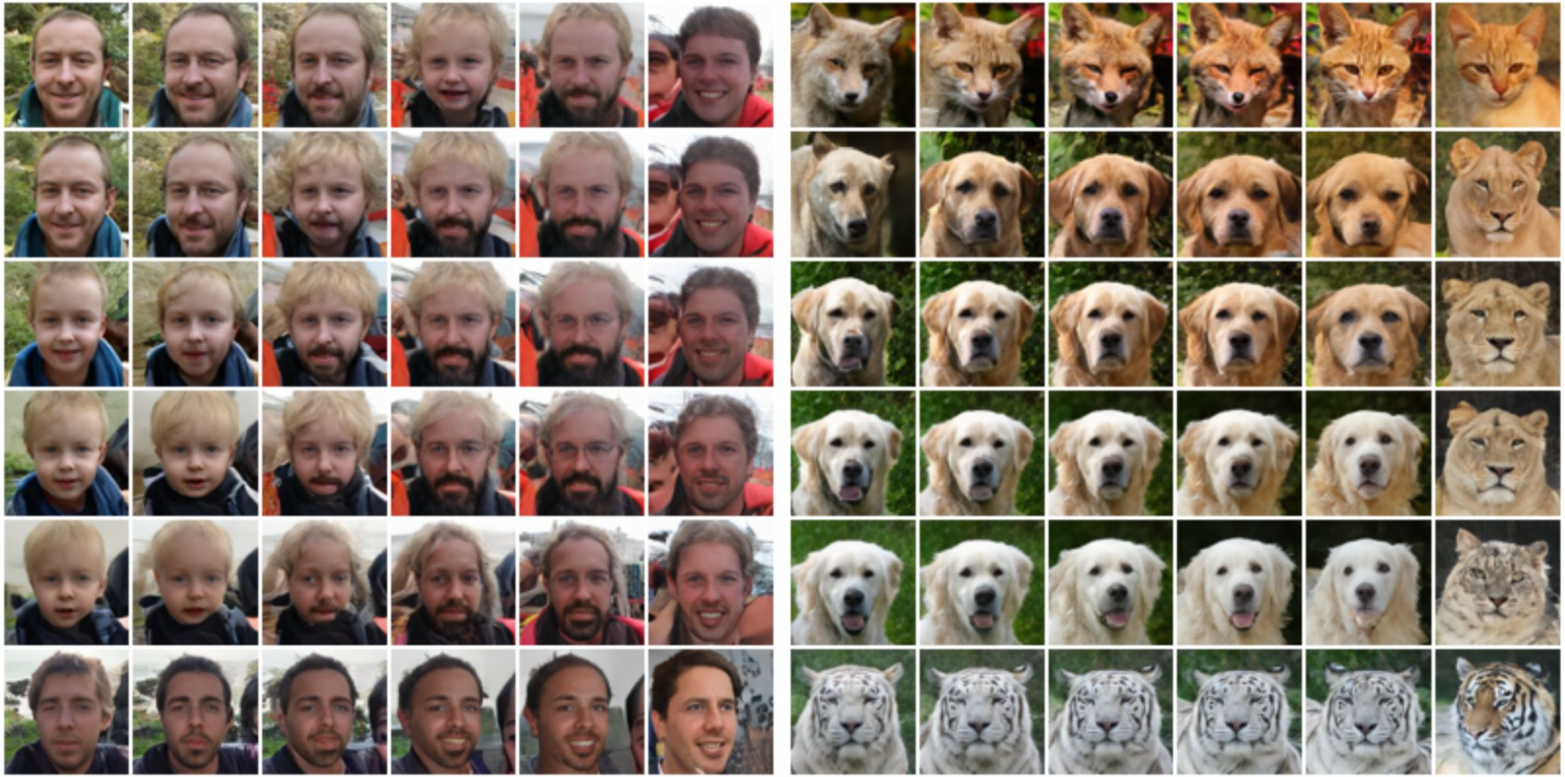
Figure 2: Generated 64 × 64 images of FFHQ with **FID=7.41065**, produced using unconditional UDPM with only 3 steps, which are equivalent to 0.3 of a single typical 64 × 64 diffusion step.

	steps	FID
DDIM	10/5	13.36/93.51
DPM-Solver	10/5	6.96/288.99
EDM	35/5	1.79/35.54
GENIE	5	11.20
DEIS	5	15.37
GGDM	5	13.77
DDGAN	2	4.08
TDPM	1	8.91
CT	1	8.70
UDPM (ours)	**<1**	**6.86**

Table 1: FID scores on the CIFAR10 dataset. UDPM uses 3 steps, which are equivalent in terms of complexity to 30% of a single denoising step used in typical diffusion models like DDPM or EDM.

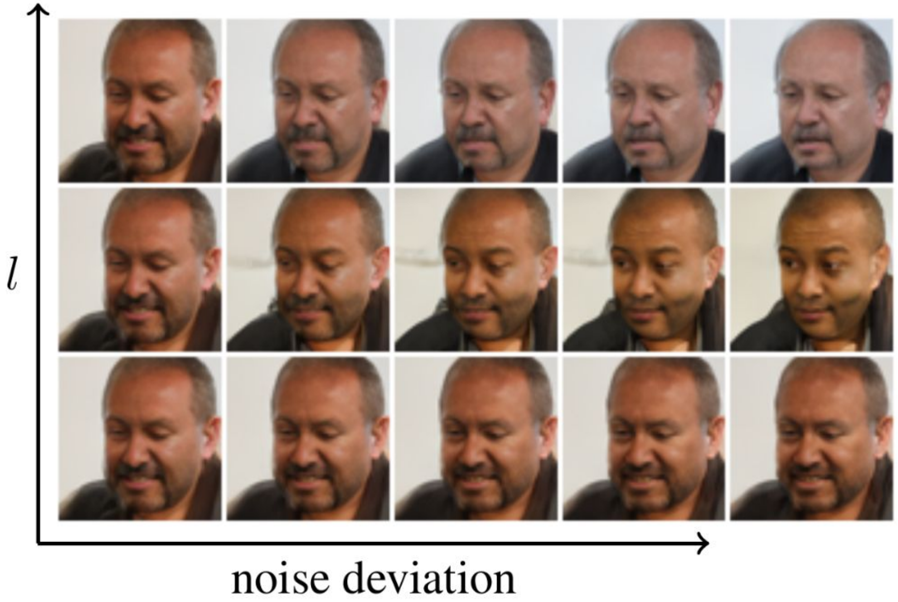
UDPM - Ablation Studies

- Latent space interpolation

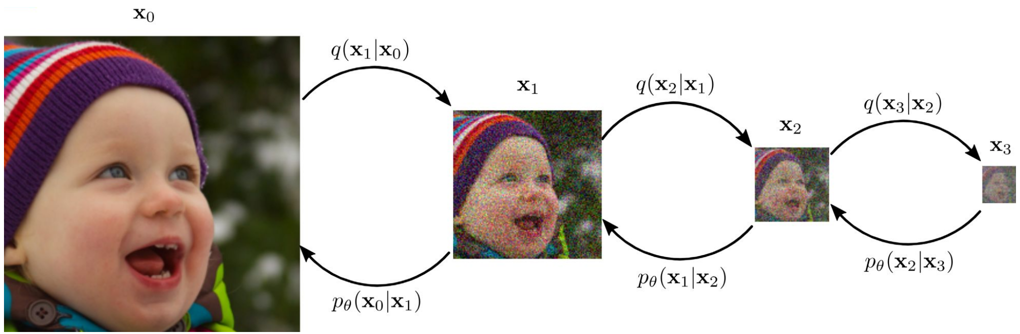


UDPM - Ablation Studies

- Single diffusion step perturbation



Upsampling Diffusion Probabilistic Models



Full Paper:

Code: