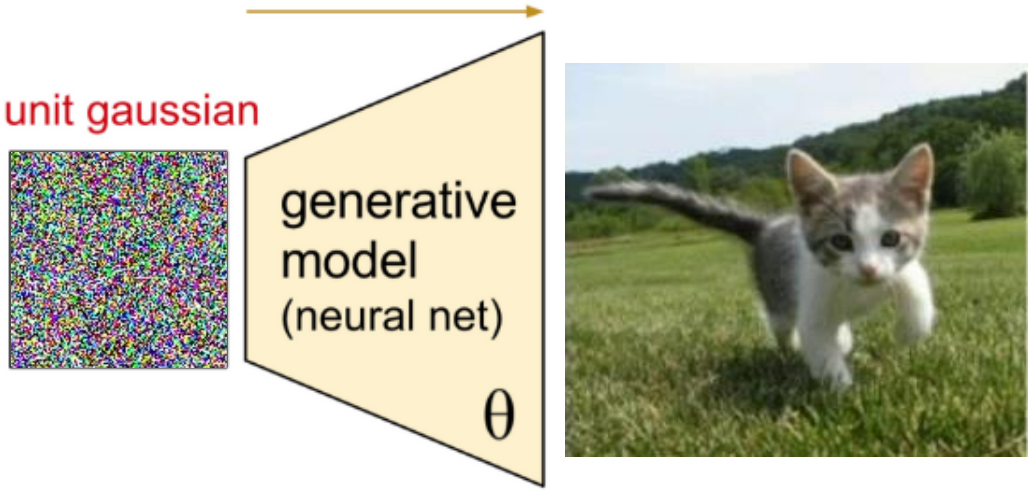
CVSP 2024 겨울 세미나



Presented By
진인환

Introduction

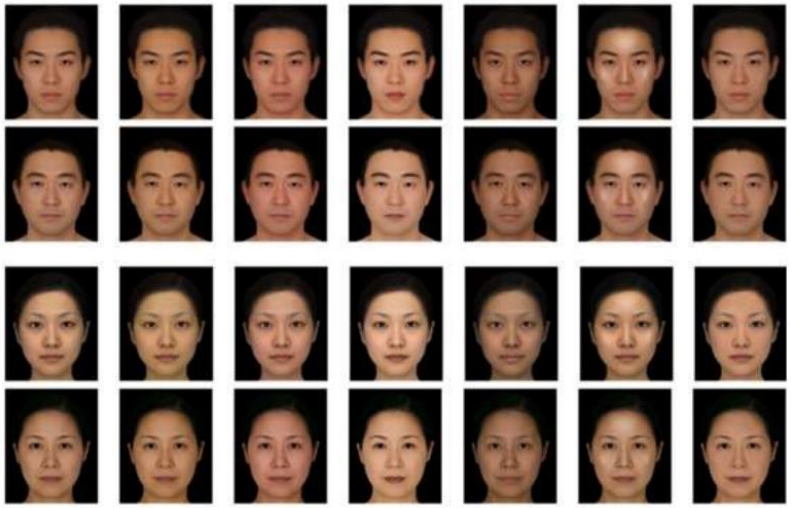
- 생성모델(Generative Model)
  - 생성모델(Generative Model)
    - 압축된 정보(noise)로부터 데이터를 생성하는 과정



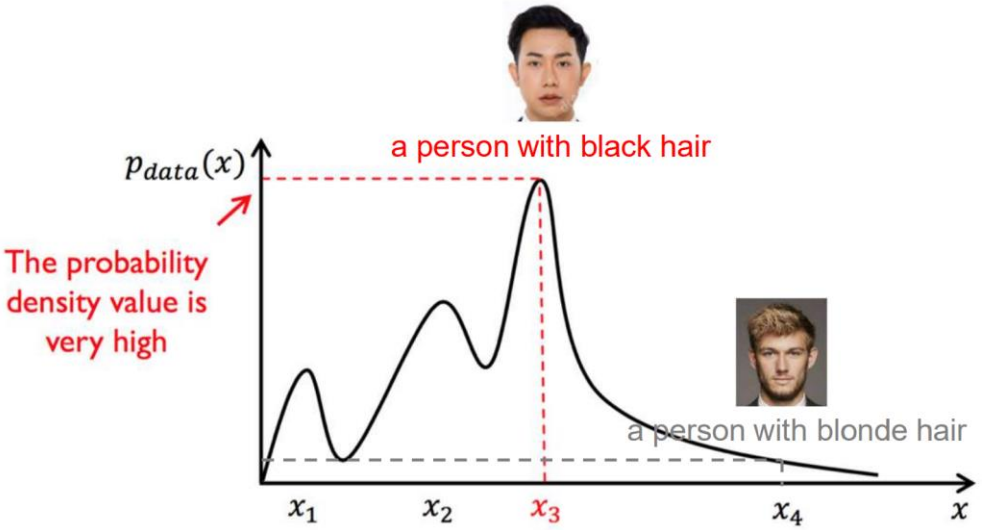
Introduction

- 생성모델(Generative Model)
  - 생성모델(Generative Model)
    - 압축된 정보(noise)로부터 데이터를 생성하는 과정

데이터셋 분포



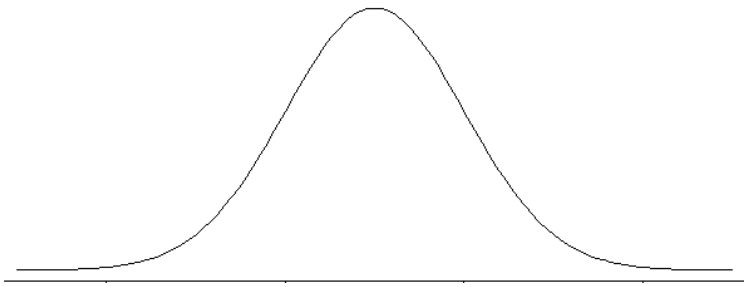
Dataset : Asian face (mostly black hair)



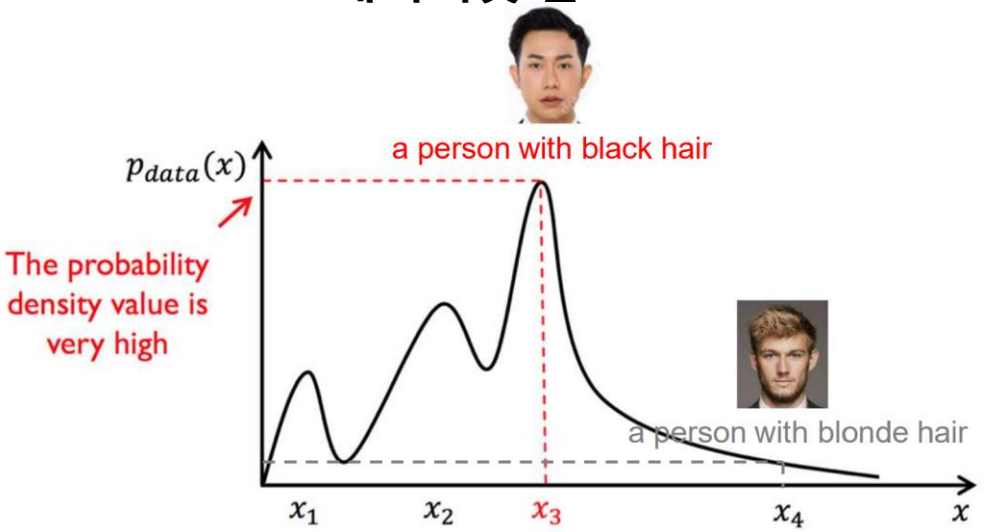
Introduction

- 생성모델(Generative Model)
  - 압축된 정보(noise)로부터 데이터를 생성하는 과정

우리가 설정한 임의의 분포



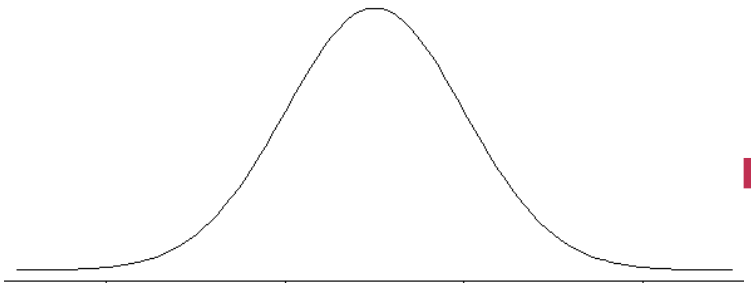
데이터셋 분포



Introduction

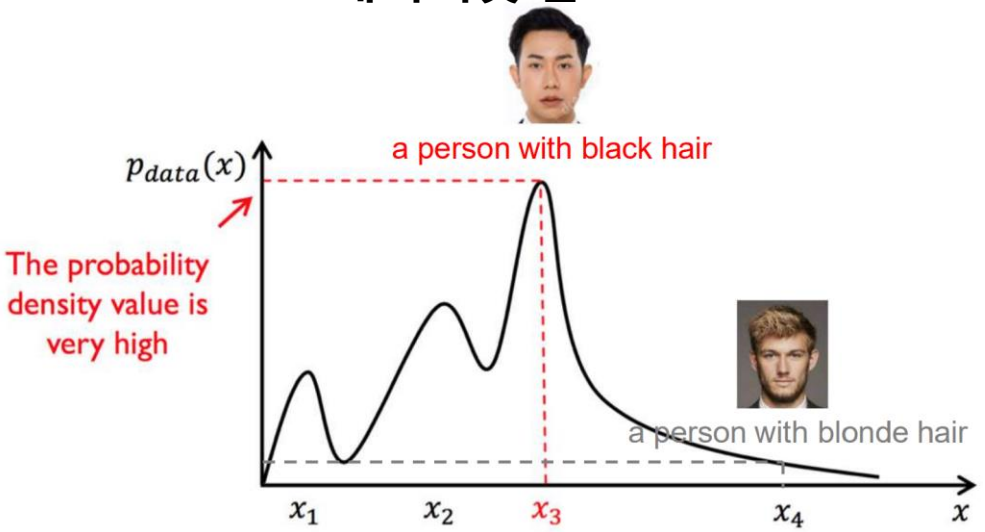
- 생성모델(Generative Model)
  - 압축된 정보(noise)로부터 데이터를 생성하는 과정

우리가 설정한 임의의 분포



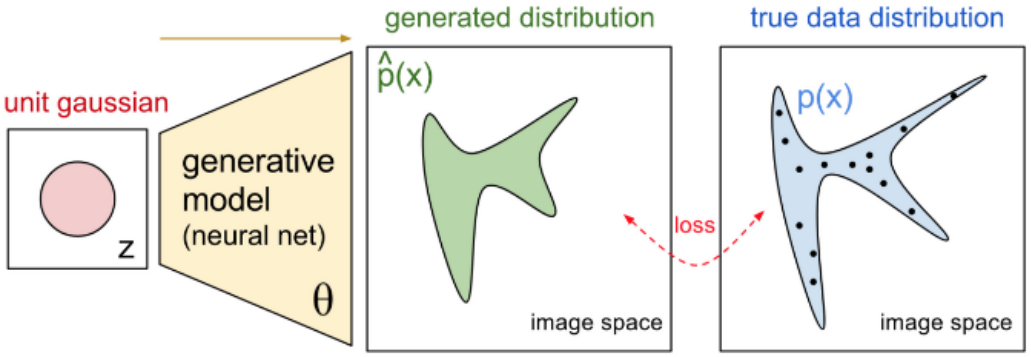
매칭

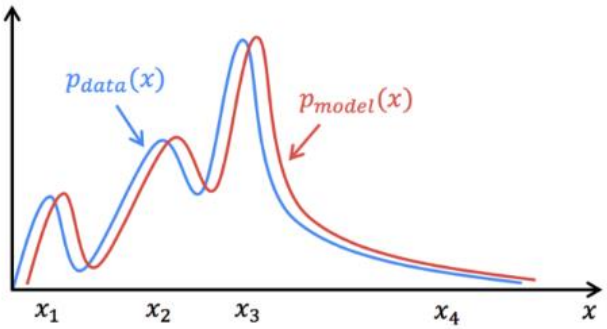
데이터셋 분포



Introduction

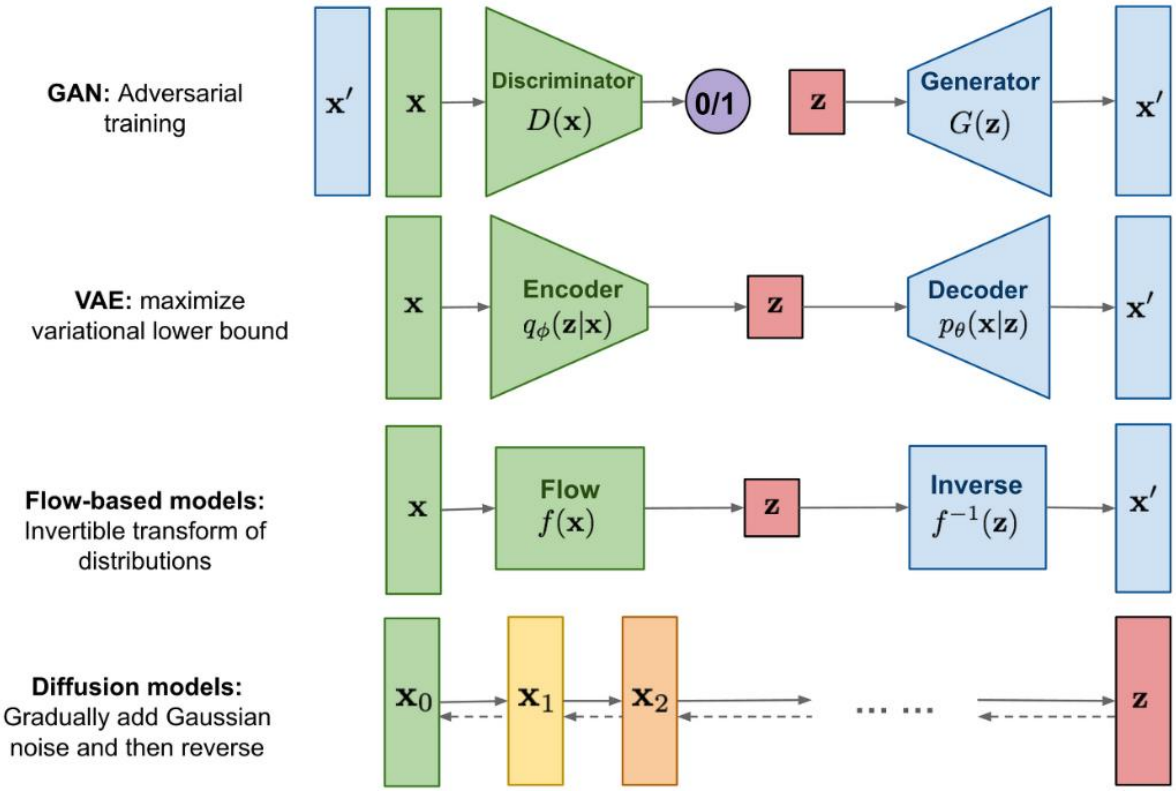
- 생성모델(Generative Model)
  - 압축된 정보(noise)로부터 데이터를 생성하는 과정
  - **우리가 설정한 임의의 분포**를 **데이터의 분포**와 잘 매칭되도록 학습하는 것이 목표





Introduction

- 생성모델(Generative Model)
  - 생성모델(Generative Model)
    - 압축된 정보(noise)로부터 데이터를 생성하는 과정
    - **우리가 설정한 임의의 분포**를 **데이터의 분포**와 잘 매칭되도록 학습하는 것이 목표



Generative Model

- Diffusion Model
  - What is diffusion
    - Forward diffusion process

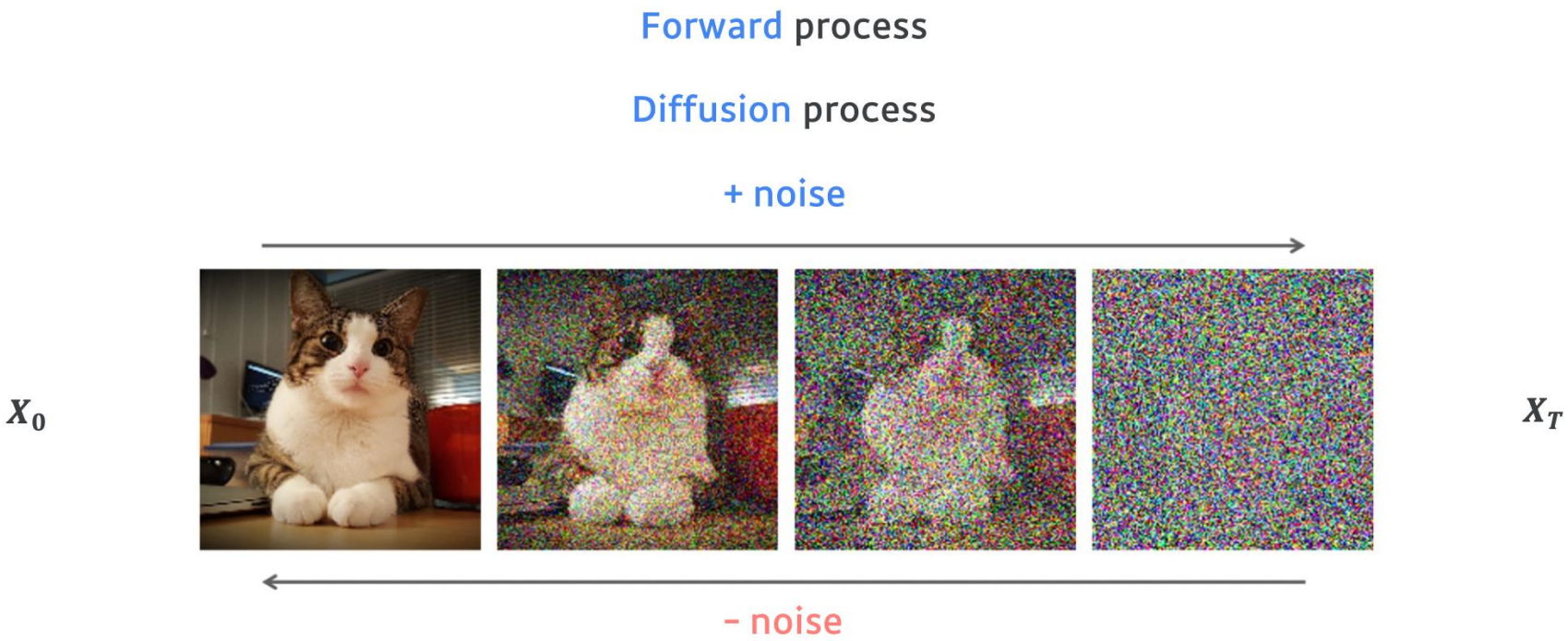


    - Reverse diffusion process



Generative Model

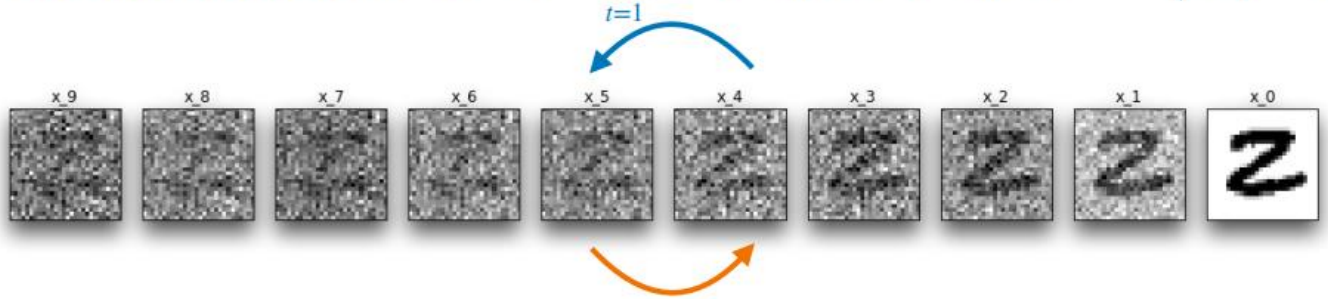
- Diffusion Model
  - What is diffusion



Generative Model

- Diffusion Model
  - Diffusion process

Forward Process $q(\mathbf{x}_{1:T} \mid \mathbf{x}_0) := \prod_{t=1}^{T} q(\mathbf{x}_t \mid \mathbf{x}_{t-1}), q(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{x}_0) := \mathcal{N}\left(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t \mathbf{I}\right)$



Posterior $q(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}\left(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t \mathbf{I}\right)$

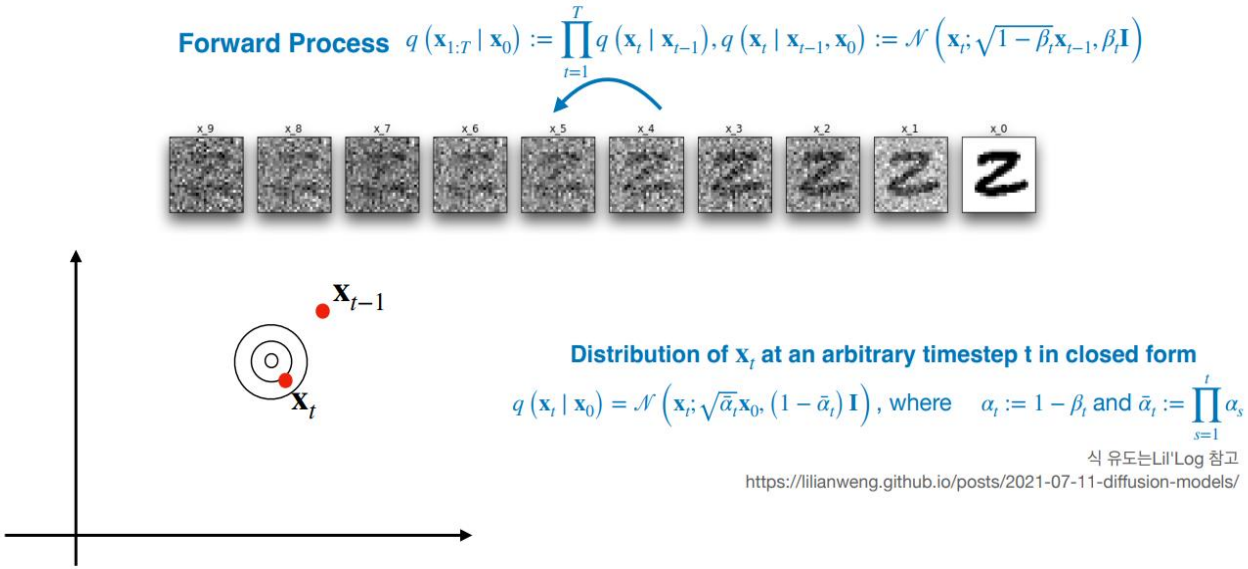
where $\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t$ and $\tilde{\beta}_t := \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$

Backward Process (Neural Networks) $p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t) := \mathcal{N}\left(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t)\right)$

Loss Function $D_{\mathrm{KL}}\left(q(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{x}_0) \| p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t)\right)$

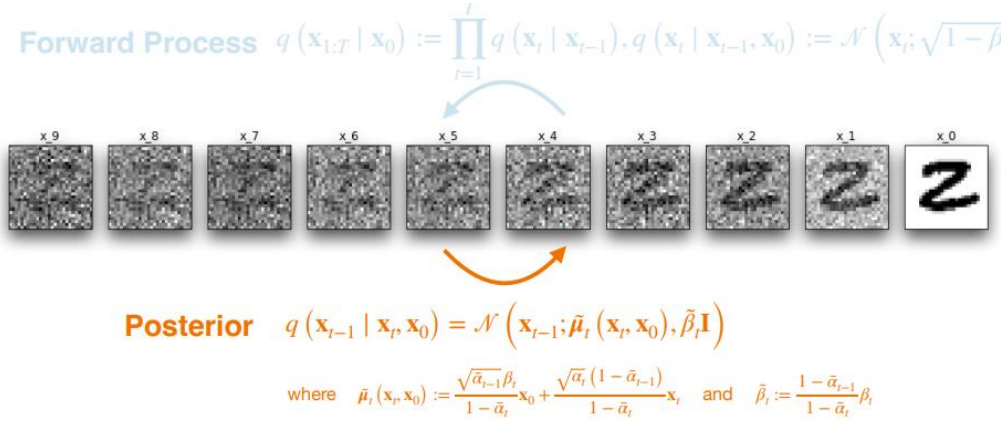
Generative Model

- Diffusion Model
  - Forward process
    - 노이즈를 추가하는 단계
      - 1부터 t시점까지 diffusion process를 진행함
      - Step이 진행됨에 따라 자연스럽게 standard normal distribution을 따르도록 설계됨
        - ✓t가 무한대에 가까워지면 mean=0, covariance = I 가 됨



Generative Model

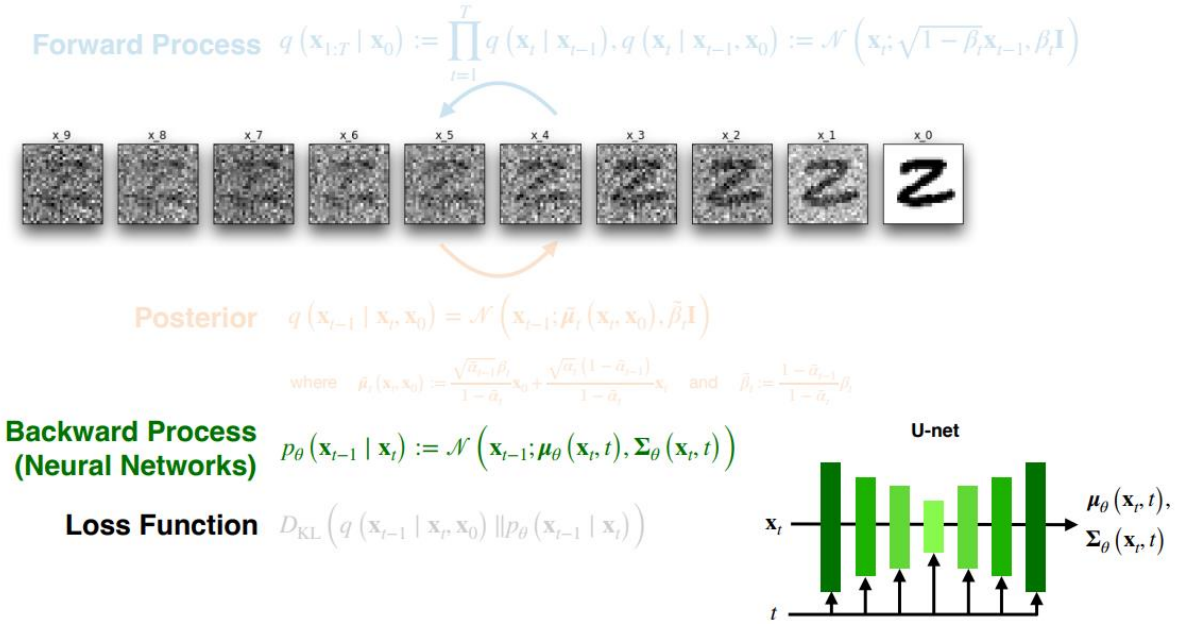
- Diffusion Model
  - Posterior
    - 정답인 X_0가 주어졌을 때 backward 방향의 분포를 구하는 단계
      - ∴ X_0과 X_t가 주어졌을 때 이전시점의 x_t-1의 확률 분포를 구함
    - Data가 주어졌을 때 backward를 구하는 수식으로 생성모델의 목표는 이전 step인 x_t만 주어졌을 때 t-1의 확률분포를 추정하도록 학습



$$q(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{x}_0) = \frac{q(\mathbf{x}_{t-1} \mid \mathbf{x}_0) q(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{x}_0)}{q(\mathbf{x}_t \mid \mathbf{x}_0)} \quad \text{by Bayes' Rule}$$

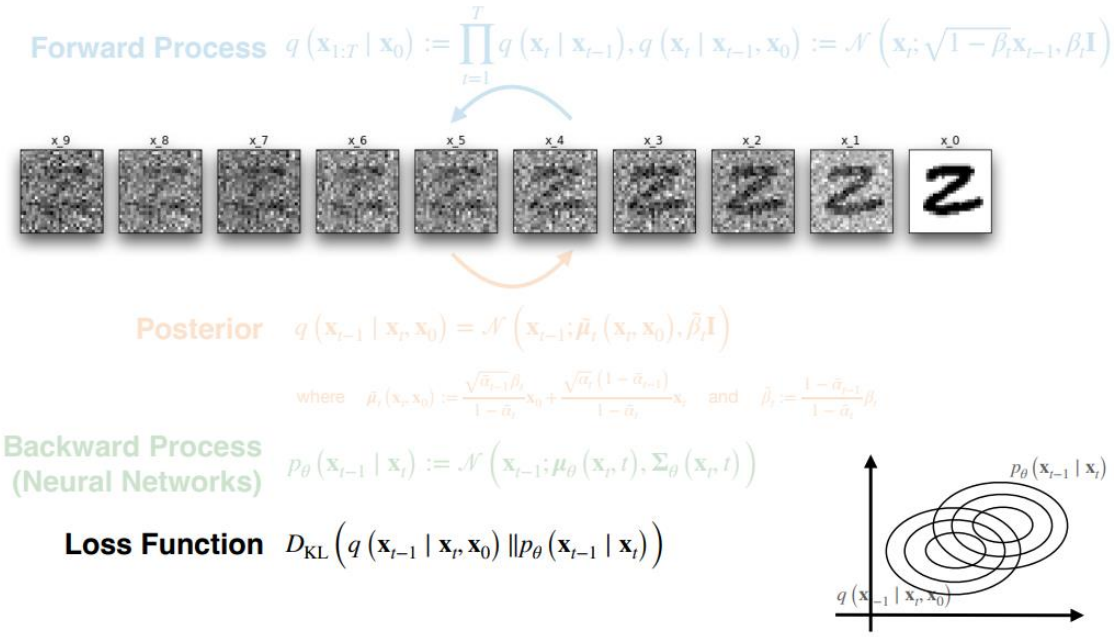
Generative Model

- Diffusion Model
  - Backward Process
    - x_t만을 통해 t-1 시점의 확률분포의 parameter을 추정하도록 neural networks를 학습
      - 분포를 Normal distribution으로 한정했을 때 mean, variance만 알아내면 됨



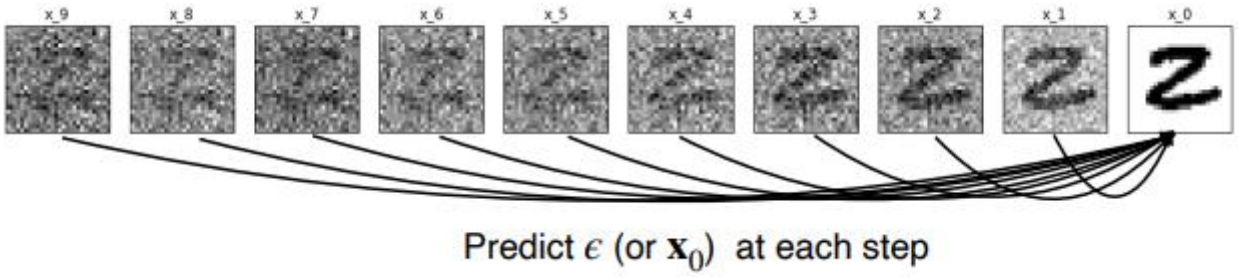
Generative Model

- Diffusion Model
  - Loss function
    - KL-divergence를 통해 posterior가 구한 분포와 network가 추정한 분포의 차이를 loss로 둠

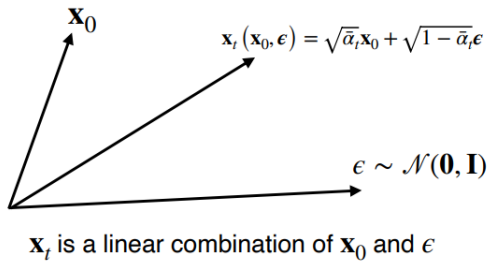


Generative Model

- DDPM
  - Diffusion model
    - Jonathan ho의 Denoising diffusion probabilistic models에서 제안
    - 어떤 시점이든 x_0를 예측하도록 학습
      - 기존의 diffusion은 t-1시점의 step의 mean과 variance를 학습함



    - X_0, X_t, ε의 관계에 주목함
      - ε : noise화 된 데이터를 만들기 위한 성분



    - 각 step에서 x_0를 예측하는 것이 아닌 ε을 학습하도록 함
      - X_0의 분포를 추정하는 것이 아닌 노이즈 성분인 ε을 추정하도록 학습함

Generative Model

- DDPM
  - Loss function
    - 분포 간의 차이가 아닌 ε을 추정하는 loss로 변경함

Loss Function $L_{\text{simple}}(\theta) := \mathbb{E}_{t,\mathbf{x}_0,\epsilon}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta\left(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t\right)\right\|^2\right]$

Generative Model

- Latent diffusion model
  - 기존의 diffusion 모델의 학습은 RGB 이미지의 높은 차원에서 진행됨
    - Computational cost가 매우 높아 memory 관점에서 매우 expensive함
  - Perceptual compression
    - Auto encoder를 통해 데이터를 down sampling하여 low dimension에서 표현 할 수 있음
  - RGB 공간의 데이터를 encode을 거쳐 latent로 인코딩하고 decoder을 통해 laten에서 image로 reconstruct함

$$z = \mathcal{E}(x),\ z \in \mathbb{R}^{h \times w \times c}$$

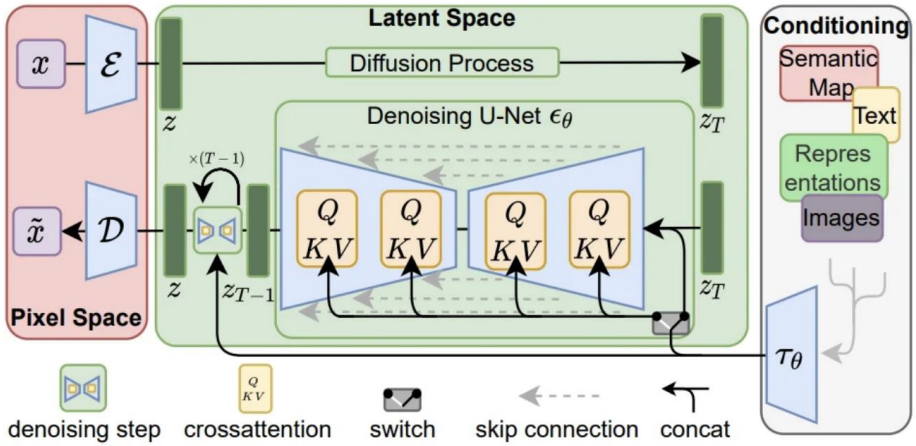
$$\tilde{x} = \mathcal{D}(z) = \mathcal{D}(\mathcal{E}(x))$$

Generative Model

- Latent diffusion model
  - Loss function
    - $\mathcal{E}$(encoder)와 $\mathcal{D}$(decoder)을 통해 compression 모델을 통해 low-dimensional latent space에 접근할 수 있음

$$L_{LDM} := \mathbb{E}_{\mathcal{E}(x),\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t)\|_2^2\right].$$

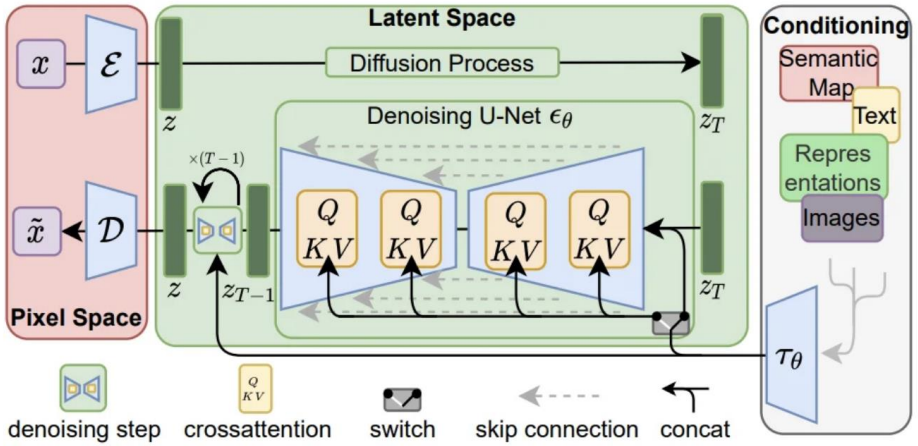
$$L_{DM} = \mathbb{E}_{x,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|_2^2\right],$$



Generative Model

- Latent diffusion model
  - Conditional denoising auto encoder를 통해 다양한 condition을 추가할 수 있음
    - -본 논문에서는 domain specific encoder Tθ을 도입함
  - Image 생성을 input y(text, semantic maps)에 따라 컨트롤 하고 image-to-image translation task가 가능해짐
    - -다양한 modality에 적응할 수 있도록 cross-attention mechanism으로 구성함

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



Generative Model

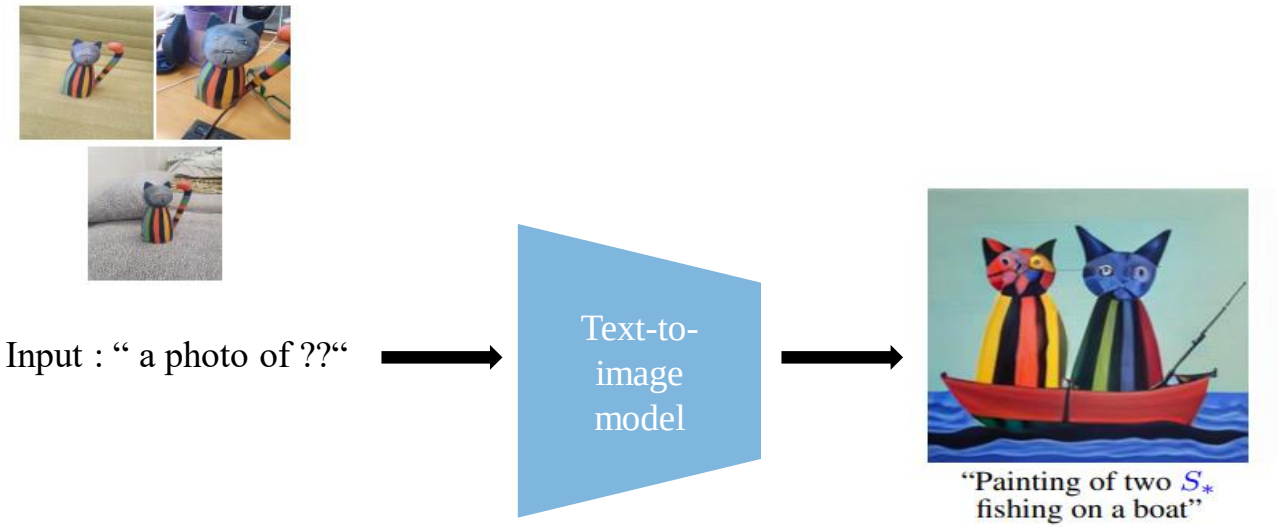
- Diffusion Model
  - DALL·E 2 vs. Imagen





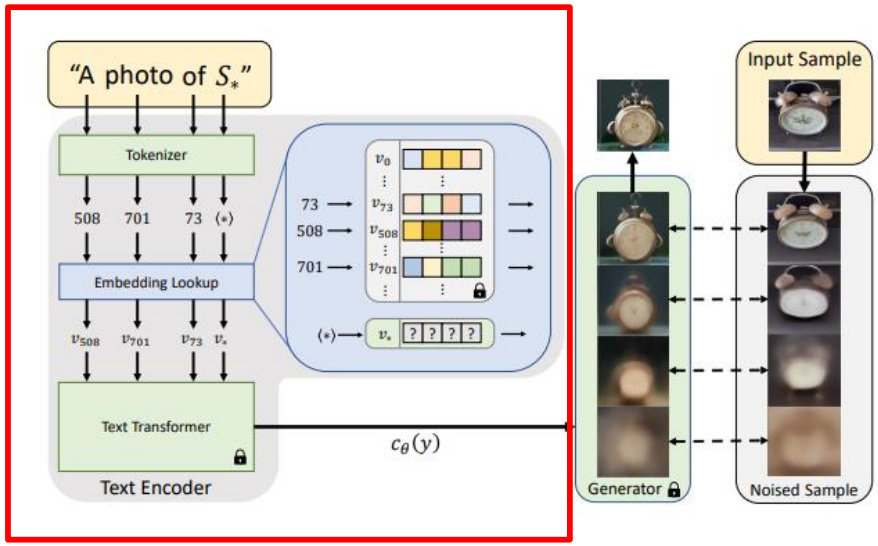
Textual inversion

- An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion
  - Introduction
    - T2I 모델들의 발전으로 다양한 장면을 생성할 수 있음
      - Text를 통해 새로운 조합과 스타일의 영상을 생성
    - 사용자가 text로 원하는 영상을 표현하기는 것에 제약이 존재함
      - 사용자의 새로운(나만의) concept을 모델에 도입하는 것은 쉽지 않음
    - Inversion?
      - Text로 부터 image를 생성하는 T2I의 반대인 Image에서 text를 뽑아내는 과정



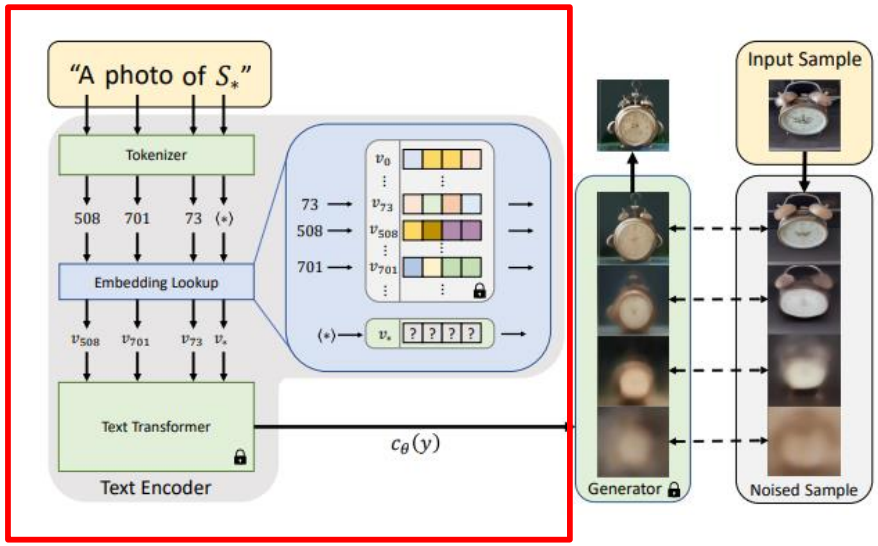
Textual inversion

- An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion
  - Introduction
    - 새로운 concept을 학습시키는 방법
      - Re-training
      - Fine tuning
      - Train transformation module
    - 기존에 학습된 embedding space에서 원하는 concept의 단어를 찾음
      - textual embedding space에서 embedding vector을 찾는 문제



Textual inversion

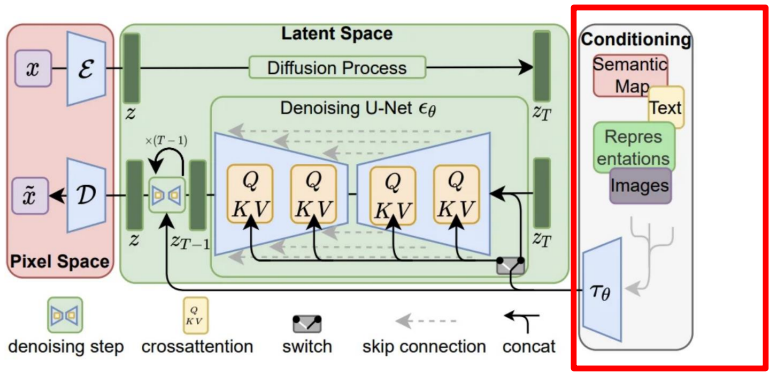
- An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion
  - Introduction
    - Contribution
      - User-provided concept에 대한 language instruction으로 personalized T2I generation 할 수 있음
      - 이 과정을 "Textual Inversion" 이라 하고 높은 수준의 의미적인 정보와 시각적인 디테일을 담은 pseudo word를 embedding space에서 찾아내고 이를 활용하여 생성함



Textual inversion

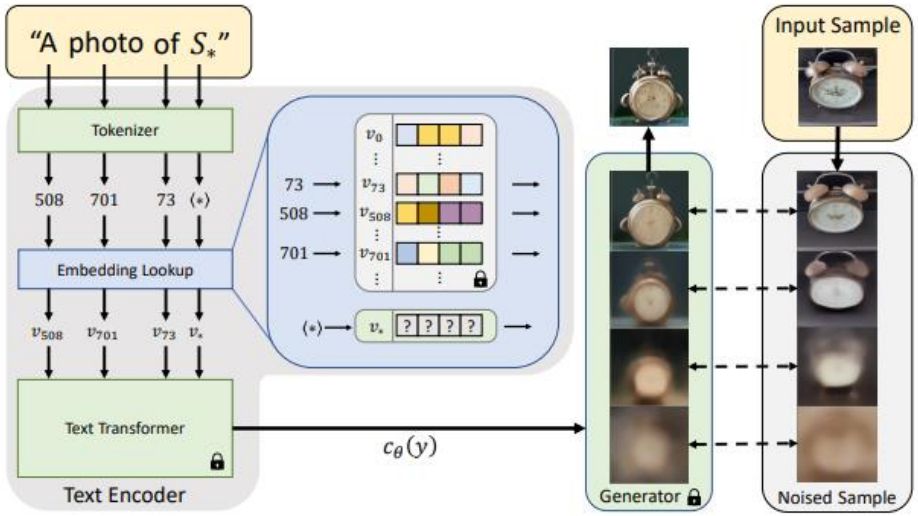
- An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion
  - Method
    - Latent diffusion model
      - Cθ는 다양한 condition y를 condition vector로 encoding 하는 모델
        - ✓Class label, segmentation mask등의 condition을 적용
      - 학습 과정에서 loss를 최적화 하기 위해 εθ와 Cθ를 joint optimize
        - ✓본 논문에서 Cθ는 bert를 활용함
      - Inference
        - ✓Random noise Z_t에서 Z_0까지 conditional denoise process을 반복적으로 수행
        - ✓Z_0을 decoder을 통과하여 image input을 얻음

$$L_{LDM} := \mathbb{E}_{z\sim\mathcal{E}(x),y,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t, c_\theta(y))\|_2^2\right],$$



Textual inversion

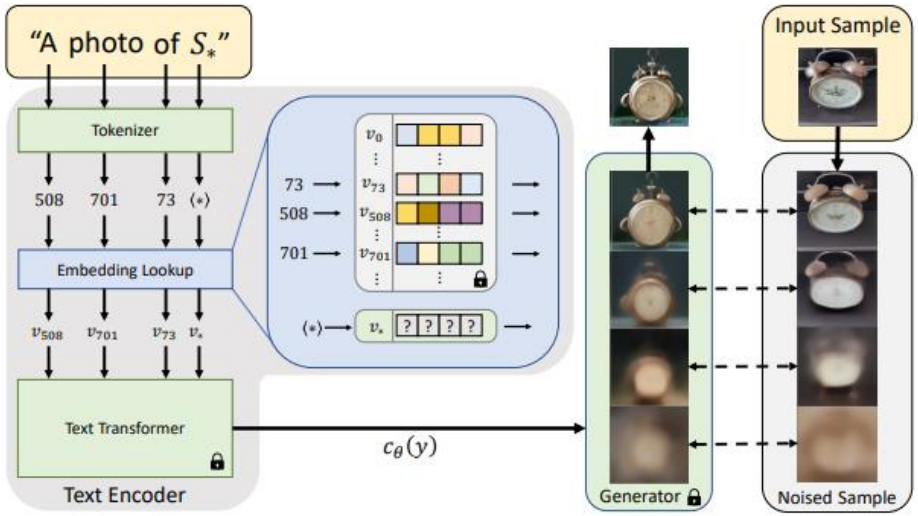
- An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion
  - Method
    - Text embedding
      - 각각의 token들은 독립된 embedding vector와 대응됨
      - Embedding vector들은 Cθ의 학습으로 얻고 해당 공간이 embedding space임
      - 본 논문은 이러한 embedding vector을 inversion의 target으로 함
        - ✓S*의 단어를 V*로 tokenizer하여 다른 단어와 함께 embedding vector을 주입하는 개념



Textual inversion

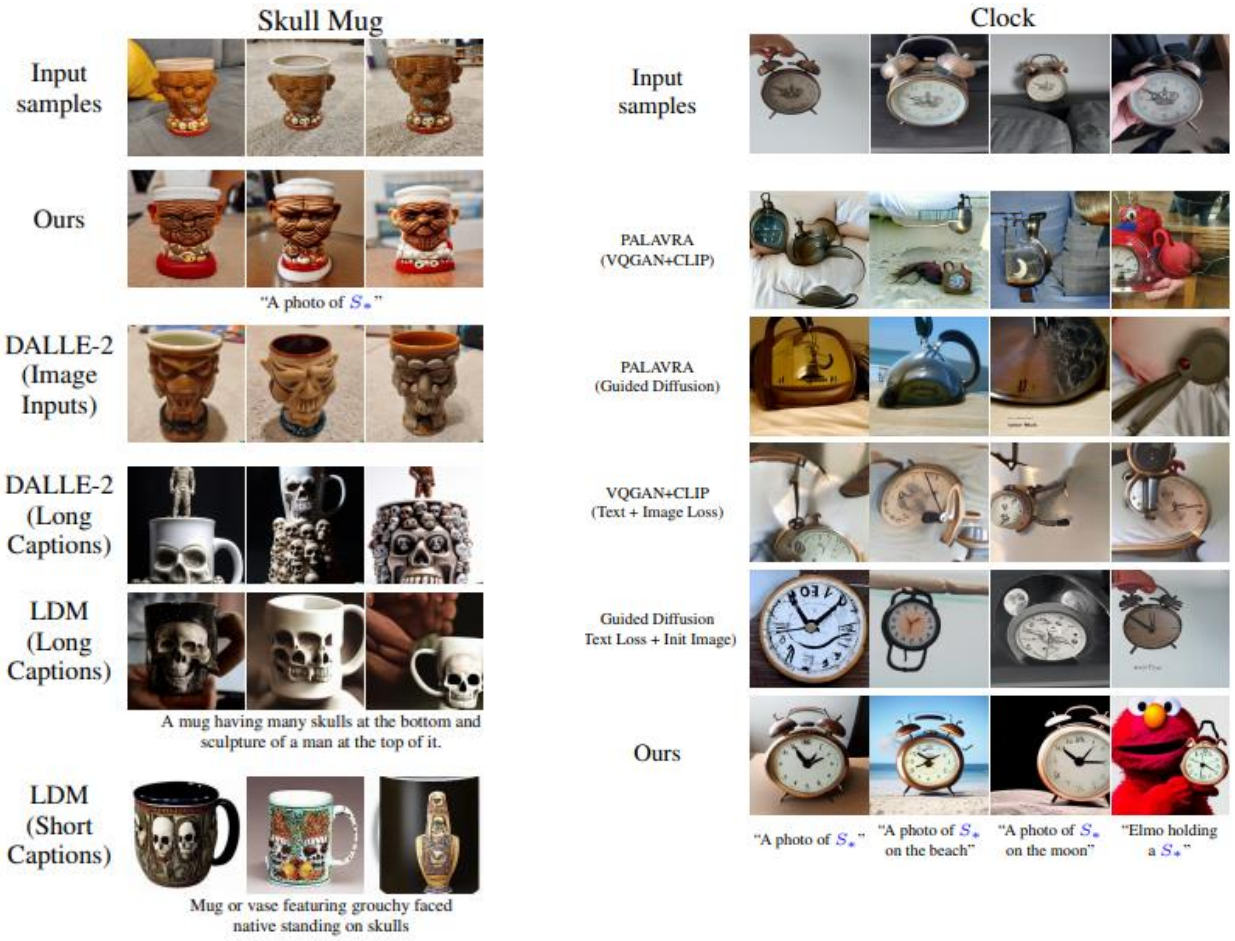
- An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion
  - Method
    - Textual inversion
      - 새로운 concept을 담은 3-5개의 input을 활용하여 V*을 direct optimize함
        - ✓Embedding vector 자체를 학습 가능
      - LDM Loss
        - ✓기존의 LDM loss를 활용하지만 $\varepsilon\theta$와 Cθ를 freeze하여 reconstruction V*을 학습시킴

$$v_* = \arg\min_v \mathbb{E}_{z\sim\mathcal{E}(x),y,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t, c_\theta(y))\|_2^2\right],$$



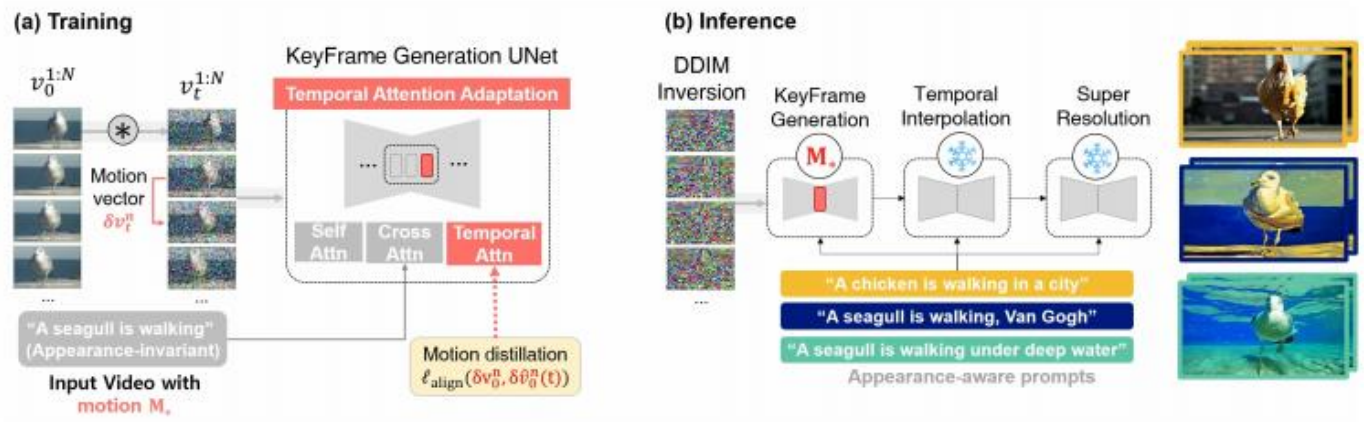
Textual inversion

- An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion
  - Qualitative Result



VMC

- VMC: Video Motion Customization using Temporal Attention Adaption for Text-to-Video Diffusion Models
  - Introduction
    - 최근 Diffusion model이 image에서 video로 확장되어 활발히 연구되고 있음
    - T2V 만으로 원하는 모션이 담긴 영상을 생성하는 것이 매우 힘듬
    - VMC를 통해 Video generation model의 customizing하여 tailored motion 비디오 생성
      - One shot tuning을 통해 reference video의 모션을 담은 비디오 생성



VMC

- VMC: Video Motion Customization using Temporal Attention Adaption for Text-to-Video Diffusion Models
  - Introduction
    - 최근 Diffusion model이 image에서 video로 확장되어 활발히 연구되고 있음
    - T2V 만으로 원하는 모션이 담긴 영상을 생성하는 것이 매우 힘듬
    - VMC를 통해 Video generation model의 customizing하여 tailored motion 비디오 생성
      - One shot tuning을 통해 reference video의 모션을 담은 비디오 생성



Input



Car + M + Underwater

- VMC: Video Motion Customization using Temporal Attention Adaption for Text-to-Video Diffusion Models
  - Methods
    - Residual frame vector 추출

✓T시점에서 latent 간의 차이에 모션 정보가 담겨있다고 가정함

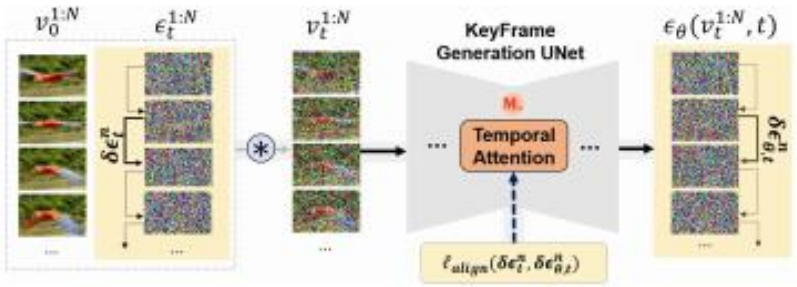
- V는 video, C는 고정된 상수로 프레임간의 간격,

$$\delta \boldsymbol{v}_t^n := \boldsymbol{v}_t^{n+c} - \boldsymbol{v}_t^n,$$

✓전체 시점에서 motion vector을 추출

- Noise level에 따라 motion vector(차)가 달라지기 때문
- DDPM의 posterior 활용

$$\begin{aligned}\delta \boldsymbol{v}_t^n &= \sqrt{\bar{\alpha}_t}(\boldsymbol{v}_0^{n+c} - \boldsymbol{v}_0^n) + \sqrt{1-\bar{\alpha}_t}(\boldsymbol{\epsilon}_t^{n+c} - \boldsymbol{\epsilon}_t^n) \\ &= \sqrt{\bar{\alpha}_t}\delta \boldsymbol{v}_0^n + \sqrt{1-\bar{\alpha}_t}\delta \boldsymbol{\epsilon}_t^n,\end{aligned}$$



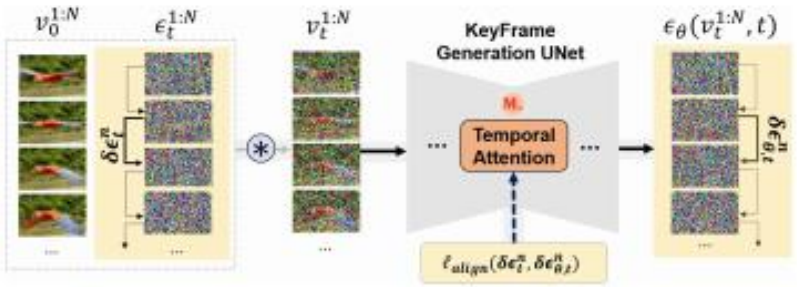
- VMC: Video Motion Customization using Temporal Attention Adaption for Text-to-Video Diffusion Models
  - Methods
    - Temporal attention layer에 motion 정보 distill
      - ✓Video diffusion model의 backward process
        - DDPM의 식 변형

$$\hat{\boldsymbol{v}}_0^{1:N}(t) := \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\boldsymbol{v}_t^{1:N} - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta(\boldsymbol{v}_t^{1:N}, t)\right)$$

      - ✓Backward process에서 motion vector의 추정
        - Denoise step에서 추정된 t 시점의 latent와 ε 를 통해 motion vector을 추정

$$\delta\hat{\boldsymbol{v}}_0^n(t) := \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\delta\boldsymbol{v}_t^n - \sqrt{1-\bar{\alpha}_t}\delta\boldsymbol{\epsilon}_{\theta,t}^n\right),$$

$$\delta\boldsymbol{\epsilon}_\theta^n(\boldsymbol{v}_t^{1:N}, t) := \boldsymbol{\epsilon}_\theta^{n+c}(\boldsymbol{v}_t^{1:N}, t) - \boldsymbol{\epsilon}_\theta^n(\boldsymbol{v}_t^{1:N}, t)$$



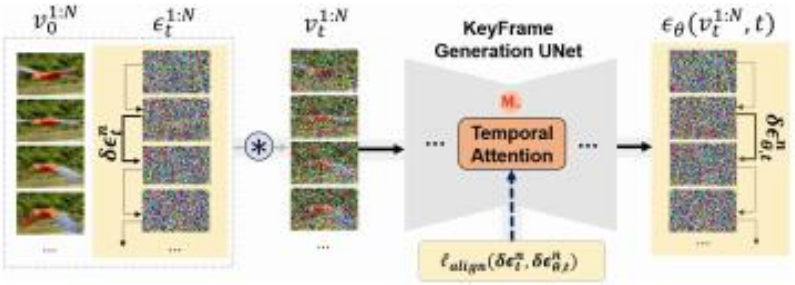
- VMC: Video Motion Customization using Temporal Attention Adaption for Text-to-Video Diffusion Models
  - Methods
    - Temporal attention layer에 motion 정보 distill
      - ✓Ground-truth motion vector

$$\delta \boldsymbol{v}_0^n = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\delta \boldsymbol{v}_t^n - \sqrt{1-\bar{\alpha}_t}\delta \boldsymbol{\epsilon}_t^n\right).$$

      - ✓Loss function
        - Motion 정보를 모델에 θ 네트워크에 distill 할 수 있음

$$\min_\theta \mathbb{E}_{t,n,\boldsymbol{\epsilon}_t^n,\boldsymbol{\epsilon}_t^{n+c}}\left[\ell_{\text{align}}\left(\delta \boldsymbol{v}_0^n, \delta \hat{\boldsymbol{v}}_0^n(t)\right)\right],$$

$$\ell_{\text{align}}\left(\delta \boldsymbol{v}_0^n, \delta \hat{\boldsymbol{v}}_0^n(t)\right) = \frac{1-\bar{\alpha}_t}{\bar{\alpha}_t}\left\|\delta \boldsymbol{\epsilon}_t^n - \delta \boldsymbol{\epsilon}_{\theta,t}^n\right\|^2.$$



VMC

- VMC: Video Motion Customization using Temporal Attention Adaption for Text-to-Video Diffusion Model
  - Qualitative result

