理論談話会 vol 2

Face2Diffusion for Fast and Editable Face Personalization

Face2Diffusionによる高速で編集可能な顔のパーソナライゼーション

スタートアップゼミ
第二回　課題
堀　秀宇

Abstract

【新規性】

1. 学習ノイズの除去による過学習の防止

   学習パイプラインから個人識別に無関係な情報を除去

2. 編集可能性の向上

   エンコードされた顔の編集可能性の向上

【有用性】

1. ID-無関係情報を除去することで過学習を防止しながら編集性をもつモデルの開発
2. 実装のしやすさと計算速度

【信頼性】

1. 従来手法より、ID忠実度・テキスト忠実度の両面で優れた性能
2. Git hubの公開とその丁寧さ

目次

1 : Introduction

研究の発展

拡散モデルの発展

拡散モデルのこれまで

年代	著者	タイトル	アプローチ	備考
2017~2019		GAN	ジェネレータとディストリビュータの2プレイヤーのミニマックスゲーム	従来生成モデル [6, 25, 37]
2020	Ho et al.	**(DDPM)Denoising Diffusion Probabilistic Models**	ノイズ除去に基づく生成モデル	拡散モデルの原点【23】
2021	Dhariwal & Nichol	**ADM（Diffusion Models Beat GANs）**	クラス分類器を使ったガイド付き拡散	高精度画像生成【14】
2021	Ho & Salimans	**Classifier-Free Guidance**	クラス情報を使わず拡散をガイド	実用性・安定性向上【22】
2021	Nichol et al.	**GLIDE**	CLIPテキスト条件付き拡散	T2Iモデル初期の代表【36】
2022	Ramesh et al.	DALL·E 2	2段階プロセス＋CLIP埋め込み	高い画像テキスト整合性【42】
2022	Saharia et al.	Imagen	T5ベースのテキスト理解＋拡散	文理解に強い高性能モデル【47】
2022	Rombach et al.	**Stable Diffusion**	潜在拡散モデル（VAE + 拡散）	軽量かつ高品質なT2Iモデル【43】

研究の発展

拡散モデルの発展

拡散モデルのこれまで

年代	著者／団体	タイトル	アプローチ	備考
2023	Rombach et al.	Stable Diffusion 2.0	潜在拡散モデルの改良版VAEの再設計、新しいテキストエンコーダ、改善されたノイズスケジュールの導入	オープンソースで公開
2023	Stability AI	Stable Diffusion XL（SDXL）	“XL”アーキテクチャによる大規模潜在拡散＆より深いU-Net構造 + CLIP-Fine-Tuning手法を採用し、テキスト／画像品質を同時に強化	768×768 → 1024×1024 超えの高解像度生成が可能　ノイズ除去能力と細部表現が大幅改善
2023	OpenAI	DALL·E 3	GPT-4系列言語理解を拡散モデルに組み込み、GPTで適切なプロンプト補完→改良型Diffusionステップを経て画像を出力	ChatGPT連携でプロンプト作成・修正を対話的にサポート　DALL·E 2よりテキスト整合性が大幅向上
2023	Saharia et al. (Google Research)	Imagen 2	大規模言語モデル（LaMDA系列）強化 + マルチスケール注意機構を組み合わせた拡散ネットワークで、文脈理解力と出力画像品質を両立	複数モダリティ統合可能で先行のImagenより精緻な画像生成
2023	Midjourney Team	Midjourney v5	独自潜在拡散ベース + 拡張カスタム・パラメータ空間で多彩なスタイルやシーンを高忠実度生成、階層型CLIPガイダンスで細部を制御	論文未公開だがコミュニティで高評価 複雑なプロンプト耐性・絵画的表現の豊かさが特徴
2023	Ho et al. (DeepMind)	DiT (Diffusion Transformer)	Transformerアーキテクチャをノイズ予測に直接適用し、ピクセル空間拡散ステップと自己注意を組み合わせたDiffusion + Transformerハイブリッド	ImageNet高解像度生成でSOTA更新 従来U-Net系より計算リソース高いが、一貫性とシャープさに優れる
2024	Rombach et al.	Stable Diffusion 2.1	Stable Diffusion 2系列のマイナーアップデート。新テキスト条件付けモジュール（改良版CLIP + 可逆正規化）で物体の細部・文字能力を強化	SD2.0の文字歪み問題を多く解消 実運用環境での安定性向上
2024	Google Research	Imagen 3	マルチモーダル事前学習を拡張し、テキストだけでなく画像中オブジェクト・構造知識を反映する拡散ネットワークを提案。高速・高解像度生成を実現	Imagen系列を超えるスケーラビリティと複雑シーン生成能力獲得 映画品質フレーム生成など応用想定
2024	Runway ML	Runway Gen-2	画像→動画拡張版拡散モデル 静止画を拡散で生成/編集 → 時系列情報考慮の2段階目拡散で自然な連続フレームを作成	動画クリップ・アニメーション生成に適用可能 従来画像拡散モデルよりTemporal Consistency重視

GAN (2017–2019)

---------- ↓拡散モデルの始まり ----------

- **DDPM (2020)** [ピクセル空間拡散モデル] **逆拡散過程でノイズを段階的**
  - **ADM (2021)** [ピクセル空間拡散モデル] **クラス分類器を用いたガイド付き拡散**
    - **Classifier-Free Guidance (2021)** [自然言語プロンプト統合系]
    - **GLIDE (2021)** [自然言語プロンプト統合系] **CLIP を使ったテキスト条件付け**
      - **DALL·E 2 (2022)** [自然言語プロンプト統合系] **2段階プロセス(CLIP + Diffusion Upsampling)**
        - **DALL·E 3 (2023)** [自然言語プロンプト統合系]
      - **Imagen (2022)** [自然言語プロンプト統合系] **T5 系の大型言語モデル**
        - **Imagen 2 (2023)** [自然言語プロンプト統合系]
          - **Imagen 3 (2024)** [マルチモーダル・高度統合系]
      - **Stable Diffusion (2022)** [潜在変数空間拡散モデル] **潜在空間拡散(Latent Diffusion)の導入**
        - **Stable Diffusion 2.0 (2023)** [潜在変数空間拡散モデル]
          - **SDXL (2023)** [潜在変数空間拡散モデル]
          - **Stable Diffusion 2.1 (2024)** [潜在変数空間拡散モデル]
        - **Midjourney v5 (2023)** [潜在変数空間拡散モデル／自然言語プロンプト統合系]
        - **Runway Gen-2 (2024)** [動画生成系] **静止画拡散 → 時系列拡散**
    - **DiT (2023)** [Transformer ハイブリッド系] **U-Net ノイズ予測 からTransformer アーキテクチャ**
      - **NVIDIA GaudiDiffusion (2024)** [文脈把握にTransformer ハイブリッド系] **局所特徴抽出にU-Net 文脈把握にTransformer**

研究の発展　Face Personalize 拡散モデルの発展

Face Personalize 拡散モデルのこれまで

年代	著者	タイトル	アプローチ	備考
2022	Gal et al.	Textual Inversion	1単語の学習で顔表現	微調整が必要【17】
2022	Ruiz et al.	DreamBooth	個人単位の全体微調整	時間・計算コスト大【46】
2023	Kumari et al.	CustomDiffusion	Cross-attention層の軽量学習	学習は速いが過学習しやすい【30】
2023	Tewel et al.	Perfusion	super-classを用いた制御	過学習を緩和【54】
2023	Gal et al.	**E4T**	事前学習済エンコーダ＋軽微調整	速く編集性も高い【18】
2023	Yuan et al.	**Celeb Basis**	有名人をベースに顔を埋め込み	identity表現に特化【63】
2023	Xiao et al.	FastComposer	局所注意により多人数混在を防止	identity fidelity 低め【60】
2023	Wei et al.	ELITE	大規模事前学習でチューニング不要	高効率・汎用性重視【59】
2023	Chen et al.	**Dream Identity**	多スケール特徴＋自己学習	テスト時チューニング不要【11】
2024	Shiohara & Yamasaki	**Face2 Diffusion (F2D)**	MSID + Expression Guidance + CGDR	identityとtextの両立を実現（本論文）

GAN (2017–2019)

---------- ↓拡散モデルの始まり ----------

- DDPM (2020)
  - ADM (2021)
    - Classifier-Free Guidance (2021)
    - GLIDE (2021)
      - DALL·E 2 (2022)
        - DALL·E 3 (2023)
      - Imagen (2022)
        - Imagen 2 (2023)
          - Imagen 3 (2024)

---------- ↓個人特徴量系拡散モデル ----------

      - Stable Diffusion (2022)
        - 直接最適化系 (Direct Optimization)
          - Textual Inversion (2022)
          - DreamBooth (2022)
          - CustomDiffusion (2023)
          - Perfusion (2023)
        - エンコーダベース最適化系 (Encoder-Based Optimization)
          - E4T (2023)
          - Celeb Basis (2023)
          - Dream Identity (2023)
          - Face2 Diffusion (2024)

GAN (2017–2019)

↓拡散モデルの始まり

- **DDPM (2020)** [ピクセル空間拡散モデル] **逆拡散過程でノイズを段階的**
  - **ADM (2021)** [ピクセル空間拡散モデル] **クラス分類器を用いたガイド付き拡散**
    - **Classifier-Free Guidance (2021)** [自然言語プロンプト統合系]
    - **GLIDE (2021)** [自然言語プロンプト統合系] **CLIP を使ったテキスト条件付け**
      - **DALL·E 2 (2022)** [自然言語プロンプト統合系] **2段階プロセス(CLIP + Diffusion Upsampling)**
        - **DALL·E 3 (2023)** [自然言語プロンプト統合系]
      - **Imagen (2022)** [自然言語プロンプト統合系] **T5 系の大型言語モデル**
        - **Imagen 2 (2023)** [自然言語プロンプト統合系]
          - **Imagen 3 (2024)** [マルチモーダル・高度統合系]
      - **Stable Diffusion (2022)** [潜在変数系拡散モデル] **潜在空間拡散(Latent Diffusion)の導入**
        - **Stable Diffusion 2.0 (2023)** [潜在変数系拡散モデル]

2 ：Back Ground

研究の背景　拡散モデルの発展

古典的な生成モデルアルゴリズム
～ VAEの発展

└── DDPM（2020）逆拡散過程でノイズを段階的

└── ADM（2021）クラス分類器を用いたガイド付き拡散

1：VAE

2：拡散モデル（Diffusion Model）

2.1：順過程（Forward Process）

2.2：逆過程（Reverse Process）

3：DDPM /ADMの成立

4：分類器不要の条件付き拡散モデル

5：自然言語プロンプト統合

6：潜在空間拡散(Diffusion Modelへの発展)

VAE が生まれるまで

古典的な生成モデル(潜在変数モデル/最尤推定・EMアルゴリズム)

分布の近似

$p_\theta(x)$ 生成モデル　　$\{x_i\}_{i=1}^N$ 観測データ（見える）　⇒　$p_{data}(x)$ 真の分布（見えない）

真の分布がわからないのでデータ集合から**経験分布**を得る

$$\hat{p}_{data}(x) = \frac{1}{N}\sum_{i=1}^{N}\delta(x - x_i)$$

**$\delta(x)$はディラックのデルタ分布

大数の法則より

$$\lim_{n\to\infty}\hat{p}_{data}^{N}(x) = p_{data}(x)$$

近似するために例えば**対数尤度最大化**を考える

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\log p_\theta(x_i) \longrightarrow \hat{\theta} = \arg\max_\theta \mathcal{L}(\theta).$$

$p_\theta(x_i)$が単一のガウス分布なら解析的に解けるが**複雑になると厳しい**

研究の背景　拡散モデルの発展

潜在変数モデルとEMアルゴリズム

例えば分布関数に観測できない変数(**潜在変数 z**)がある場合

$$p_\theta(x) = \int p_\theta(x,z)\,dz = \int p_\theta(x \mid z)\,p_\theta(z)\,dz.$$

$p_\theta(z)$は潜在変数の事前分布(これは目的に応じて仮定を行う)**

** $p_\theta(x \mid z)$は潜在変数zから観測xが生成する条件付き確率

$$\mathcal{L}(\theta) = \sum_{i=1}^{N} \log p_\theta(x_i) = \sum_{i=1}^{N} \log\left(\int p_\theta(x_i,z)\,dz\right)$$

積分を閉形式で解くことはできないので

エビデンス下限値を定義　**EMアルゴリズム**

によって**下限値(ELBO)を最大化**する事で実質的に

尤度も最大化させる

エビデンス下限値（ELBO）の導出(Jensenの不等式)

$$\log p_\theta(x) = \log \int p_\theta(x,z)\,dz = \log \int p_\theta(z)\,p_\theta(x \mid z)\,dz$$

**zは潜在変数 , xは観測データ

潜在変数の近似分布関数 $q(z)$ を導入

$$\log p_\theta(x) = \log \int q(z)\,\frac{p_\theta(x,z)}{q(z)}\,dz = \log\, \mathbb{E}_{z\sim q}\left[\frac{p_\theta(x,z)}{q(z)}\right]$$

Jensenの不等式より

$$\log\, \mathbb{E}_{z\sim q}\left[\frac{p_\theta(x,z)}{q(z)}\right] \geq \mathbb{E}_{z\sim q}\left[\log\left(\frac{p_\theta(x,z)}{q(z)}\right)\right] = \mathcal{L}(q,\theta;x)$$

よって**エビデンス下限値(ELBO)** $\mathcal{L}(q,\theta;x)$ が導出された

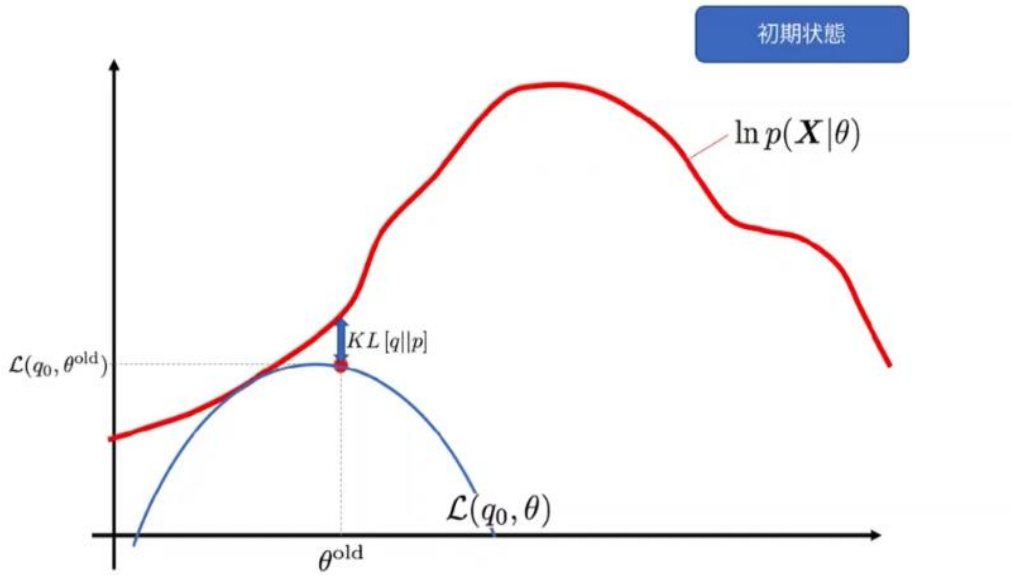
研究の背景　拡散モデルの発展

EMアルゴリズムによるELBO項の最大化

$$\mathcal{L}(q,\theta;x) = \int q(z)\,\log\frac{p_\theta(x,z)}{q(z)}\,dz$$

$$= \int q(z)\,\log p_\theta(x,z)\,dz \;-\; \int q(z)\,\log q(z)\,dz.$$

$$= \mathbb{E}_{z\sim q}\big[\log p_\theta(x,z)\big] \;-\; \mathbb{E}_{z\sim q}\big[\log q(z)\big].$$



Eステップ（Expectation Step）

$\mathcal{L}(q,\theta;x)$　を$q(z)$について最大化するステップ

等号成立条件で**周辺対数尤度** $\log p_\theta(x)$ が一致(最大化)されるので

$$q(z) = p_\theta(z\mid x) \quad\Longrightarrow\quad \mathcal{L}(q,\theta;x) = \log p_\theta(x).$$

等号成立条件である**真の潜在変数事後分布$p_\theta(z\mid x)$**はわからないので

$$q^{(t)}(z) = p_{\theta^{(t)}}(z\mid x)$$

更新パラメータ $\theta^{(t)}$の下で事後分布を計算

Mステップ（Maximization Step）

$$\mathcal{L}\big(q^{(t)},\theta;x\big) = \int p_{\theta^{(t)}}(z\mid x)\,\log p_\theta(x,z)\,dz \;-\; \int p_{\theta^{(t)}}(z\mid x)\,\log p_{\theta^{(t)}}(z\mid x)\,dz.$$

$q^{(t)}(z)$の分布を固定して$\theta^{(t)} \to \theta^{(t+1)}$に更新するので**下線部**は定数

$$\theta^{(t+1)} = \arg\max_\theta \int p_{\theta^{(t)}}(z\mid x)\,\log p_\theta(x,z)\,dz = \arg\max_\theta \underbrace{\mathbb{E}_{z\sim p_{\theta^{(t)}}(z|x)}\big[\log p_\theta(x,z)\big]}_{Q(\theta,\theta^{(t)})}.$$

研究の背景　拡散モデルの発展

EMアルゴリズムの限界とVAE

限界１：局所最適解への収束
　　大域的最適解への収束は**初期値**に依存
限界２：**事後分布の計算**が困難な場合(Eステップ)
限界３：解析的更新式による**最適化の限界**(Mステップ)
　　勾配法で最適化を行う必要性がある

Eステップの限界

$$q^{(t)}(z) = p_{\theta^{(t)}}(z \mid x)$$

潜在変数が**連続高次元**で**事後分布がNN**の時など

Mステップの限界

$$\theta^{(t+1)} = \arg\max_{\theta} \underbrace{\mathbb{E}_{z\sim p_{\theta^{(t)}}(z|x)}[\log p_\theta(x,z)]}_{Q(\theta,\theta^{(t)})}.$$

交互最適化で毎回**勾配法**を用いると計算コストが高い

VAE (variational Autoencoder)

Amortized 変分推論　（例えばガウス分布など）

$$q^{(t)}(z) = p_{\theta^{(t)}}(z \mid x) \quad \Longrightarrow \quad q_\phi(z \mid x) = \mathcal{N}\big(z;\, \mu_\phi(x),\, \mathrm{diag}(\sigma_\phi(x)^2)\big).$$

$\boldsymbol{\theta^{(t)}}$毎に更新が必要な潜在変数の近似分布$\boldsymbol{q^{(t)}(z)}$ではなく
一度の $\boldsymbol{\phi}$学習だけで良いEncoder Network $\boldsymbol{q_\phi(z\,|x)}$ に置き換える

DNNによる潜在変数モデルの表現(深層潜在変数モデル)

$$\mathcal{L}(\theta,\phi;\mathcal{X}) = \sum_{i=1}^{N}\Big[\underbrace{\mathbb{E}_{z\sim q_\phi(z|x^{(i)})}\big[\log p_\theta(x^{(i)},z)\big]}_{} - \mathbb{E}_{z\sim q_\phi(z|x^{(i)})}\big[\log q_\phi(z \mid x^{(i)})\big]\Big].$$

$\boldsymbol{\theta}$と$\boldsymbol{\phi}$を同時に最適化(更新)できる目的関数 $\mathcal{L}(\theta,\phi;\mathcal{X})$ を設定

再パラメータ化トリックによる直接的勾配学習

$$z = \mu_\phi(x) + \sigma_\phi(x)\,\epsilon, \quad \epsilon \sim \mathcal{N}(0,I)$$

ガウスノイズ $\boldsymbol{\varepsilon \sim \mathcal{N}(0,I)}$を導入し潜在変数 $\boldsymbol{z}$ を置き換え(再パラメータ化)

$$\underbrace{\mathbb{E}_{z\sim q_\phi(z|x)}\big[\log p_\theta(x \mid z)\big]}_{} = \mathbb{E}_{\epsilon\sim\mathcal{N}(0,I)}\Big[\log p_\theta\big(x \mid \mu_\phi(x)+\sigma_\phi(x)\,\epsilon\big)\Big].$$

$\boldsymbol{\theta}$と$\boldsymbol{\phi}$の両方で微分可能な形に...　　**自動微分を用いた勾配更新**を行う

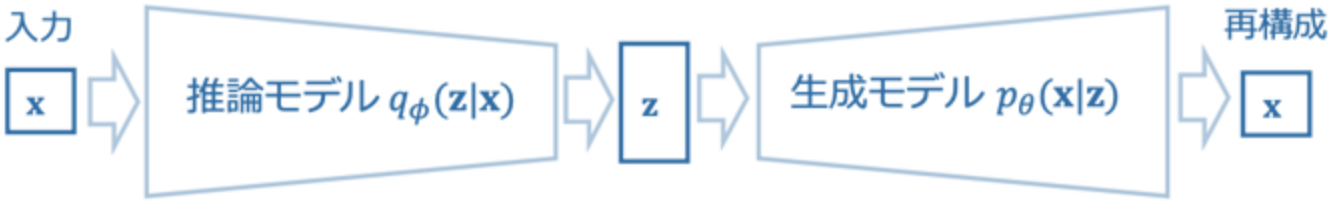
研究の背景　拡散モデルの発展

VAEの発展
~拡散モデルの概念の登場

└ DDPM（2020）逆拡散過程でノイズを段階的

└ ADM（2021）クラス分類器を用いたガイド付き拡散

VAEの全体像



拡散モデル（Diffusion Model）

2.1：階層型VAE

・複雑なデータは**単一の解像度**で潜在空間の情報を捉えきれない
・入力データの**グローバル & ローカル**な特徴を表現したい

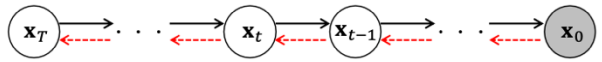
$$q_{\phi_1}(z_1 \mid x) = \mathcal{N}\Big(z_1;\, \mu_{1,\phi}(x),\, \mathrm{diag}\big(\sigma_{1,\phi}(x)^2\big)\Big),$$ **下位潜在z_1をデータxから推論**

$$q_{\phi_2}(z_2 \mid z_1) = \mathcal{N}\Big(z_2;\, \mu_{2,\phi}(z_1),\, \mathrm{diag}\big(\sigma_{2,\phi}(z_1)^2\big)\Big).$$ **上位潜在z_2を下位潜在z_1から推論**

$$q_\phi(z_2,\, z_1 \mid x) = q_{\phi_1}(z_1 \mid x)\; q_{\phi_2}(z_2 \mid z_1)$$ **組み合わせて真の事後分布に近似**

この**階層型VAE**の一般形が**マルコフ連鎖型 VAE**

$$q(x_{t+1} \mid x_t) = \mathcal{N}\Big(x_{t+1};\, \sqrt{1-\beta_{t+1}}\, x_t,\, \beta_{t+1} I\Big)$$



**x_0は観測変数　$x_1 \cdots x_T$は潜在変数

研究の背景　拡散モデルの発展

2.2：順過程(拡散過程)

各階層ごとの遷移で**ガウスノイズ**を少しずつ足す

Step1： x_0(元画像)を最下層の潜在変数として学習

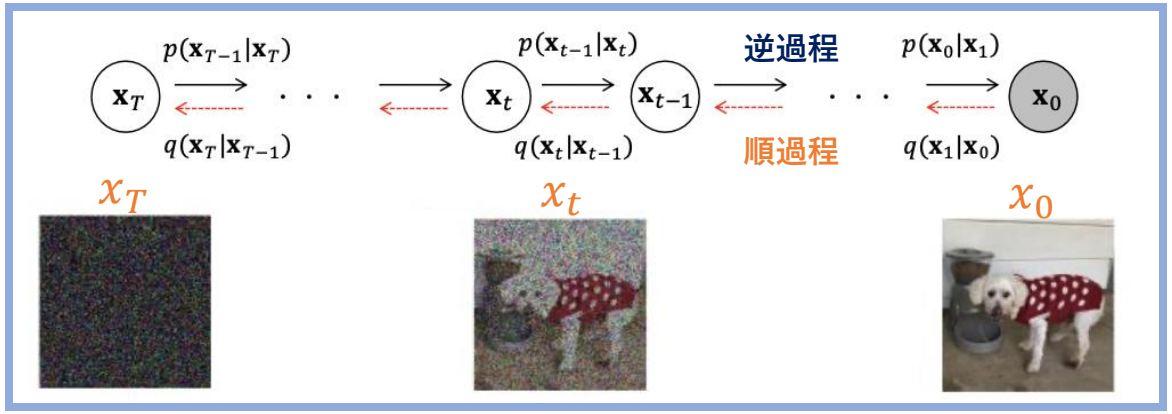
Step2： $x_1 \cdots x_T$の各ステップでガウスノイズを付与

潜在変数：すでにノイズが足された一つ前の画像

潜在変数分布：一つ前の潜在変数との条件付き確率 → ガウスノイズ

$$q(x_{t+1} \mid x_t) = \mathcal{N}\left(x_{t+1};\ \sqrt{1-\beta_{t+1}}\,x_t,\ \beta_{t+1}I\right)$$ **β_{t+1}はノイズ率(回数に反比例)

この操作を繰り返す事で最終的に**純粋なガウスノイズ** $\mathcal{N}(0, I)$ に近づく



2.3：逆過程(生成過程)

逆過程の各ステップをガウスで近似

$$p_\theta\left(x_t \mid x_{t+1}, t+1\right) = \mathcal{N}\left(x_t;\ \mu_\theta(x_{t+1}, t+1),\ \Sigma_\theta(t+1)\right).$$

**$\mu_\theta(x_{t+1}, t+1)$はNNで出力される再構成平均，$\Sigma_\theta(t+1)$ はステップ毎に固定した分散

再構成平均 μ_θ の導出

拡散過程からベイズの定理で計算される**真の生成過程**は

$$\tilde{\mu}_{t+1}(x_{t+1}, x_0) = \frac{1}{\sqrt{\alpha_{t+1}}}\left(x_{t+1} - \frac{\beta_{t+1}}{\sqrt{1-\bar{\alpha}_{t+1}}}x_0\right),$$

しかし逆過程では**元画像 x_0** が分からないため以下の方法で近似

再パラメータ化トリックを用いて**ガウスノイズ**に置き換え

$$x_{t+1} = \sqrt{\bar{\alpha}_{t+1}}\,x_0 + \sqrt{1-\bar{\alpha}_{t+1}}\,\epsilon, \quad \epsilon \sim \mathcal{N}(0, I).$$

$$\mu_\theta\left(x_{t+1}, t+1\right) = \frac{1}{\sqrt{\alpha_{t+1}}}\left(x_{t+1} - \frac{\beta_{t+1}}{\sqrt{1-\bar{\alpha}_{t+1}}}\,\epsilon_\theta(x_{t+1}, t+1)\right)$$

**βはステップ t でどれだけのノイズを付与するかの”ノイズ分散率”，
**αはステップ t で前のステップ $t-1$ の情報を残すかの“信号残存率”

ノイズ予測NN ε_θでノイズを予測する仕組みを導入　>> **DDPM**

研究の背景　拡散モデルの発展

VAEの発展
~拡散モデルの概念の登場
GAN (2017–2019)
└── DDPM (2020) 逆拡散過程でノイズを段階的
　　└── ADM (2021) クラス分類器を用いたガイド付き拡散

1 ：VAE

2 ：拡散モデル（Diffusion Model）

2.1：階層型VAE

2.2：順過程（Forward Process）

2.3：逆過程（Reverse Process）

3 ：DDPM / ADM の成立

4 ：分類器不要の条件付き拡散モデル

5 ：自然言語プロンプト統合

6 ：潜在空間拡散(Diffusion Modelへの発展)

DDPMの成立(ノイズ予測NN ε_θで直接近似)

Denoising Diffusion Probabilistic Model

損失関数の定義

真のノイズ ε とノイズ予測NN ε_θの**二乗誤差の最小化**

$$L_{\text{simple}} = \mathbb{E}_{x_0,\,\epsilon,\,t}\left[\left\|\epsilon - \epsilon_\theta(x_t,\,t)\right\|^2\right], \quad x_t = \sqrt{\bar{\alpha}_t}\,x_0 + \sqrt{1-\bar{\alpha}_t}\,\epsilon.$$

これは生成モデルの周辺対数尤度 $\log p_\theta(x_0)$ の最大化と等しい

** 厳密にはKLダイバージェンスの最小化と周辺対数尤度の最大化が等価

$$\log p_\theta(x_0) \geq \mathbb{E}_{q(x_{1:T}|x_0)}\left[\log p(x_T) + \sum_{t=1}^{T} \log p_\theta(x_{t-1} \mid x_t) - \sum_{t=1}^{T} \log q(x_t \mid x_{t-1})\right]$$

$$= -D_{\mathrm{KL}}\big(q(x_T \mid x_0)\,\|\,p(x_T)\big)$$

$$- \sum_{t=2}^{T} \mathbb{E}_{q(x_{t-1},x_t|x_0)}\left[D_{\mathrm{KL}}\big(q(x_{t-1} \mid x_t, x_0)\,\|\,p_\theta(x_{t-1} \mid x_t)\big)\right]$$

$$- \mathbb{E}_{q(x_1|x_0)}\left[-\log p_\theta(x_0 \mid x_1)\right]$$

$$D_{\mathrm{KL}}\big(q(x_{t-1} \mid x_t, x_0)\,\|\,p_\theta(x_{t-1} \mid x_t)\big) \approx \frac{1}{2\beta_t}\left\|\tilde{\mu}_t - \mu_\theta(x_t,t)\right\|^2 \propto \left\|\epsilon - \epsilon_\theta(x_t,t)\right\|^2$$

研究の背景　拡散モデルの発展

VAEの発展
~拡散モデルの概念の登場
GAN (2017–2019)
└ DDPM (2020) 逆拡散過程でノイズを段階的
　└ ADM (2021) クラス分類器を用いたガイド付き拡散

1：VAE

2：拡散モデル（Diffusion Model）

2.1：階層型VAE

2.2：順過程（Forward Process）

2.3：逆過程（Reverse Process）

3：DDPM / ADM の成立

4：分類器不要の条件付き拡散モデル

5：自然言語プロンプト統合

6：潜在空間拡散(Diffusion Modelへの発展)

ADMの成立　（クラス分類器を用いた「ガイド付き拡散」）

Classifier-Guided Diffusion

学習済みクラス分類器 $p_\phi(y \mid x_t, t)$ をノイズ過程に合わせて用意

$p_\phi(y \mid x_t, t)$：**ノイズのある画像 x_t** が**元画像のクラスラベル y** に属する確率

クラス分類器対数確率 $\log p_\phi(y \mid x_t, t)$ の勾配で制御

DDPMの各ステップにおける逆過程サンプリング

$$x_{t-1} = \mu_\theta(x_t, t) + \sqrt{\beta_t}\, z, \quad z \sim \mathcal{N}(0, I).$$

ADMの各ステップにおける逆過程サンプリング

$$x_{t-1} = \mu_\theta(x_t, t) + s\,\beta_t \nabla_{x_t} \log p_\phi(y \mid x_t, t) + \sqrt{\beta_t}\, z, \quad z \sim \mathcal{N}(0, I).$$

元の無条件モデルはそのまま使い
クラス分類器の勾配だけをサンプリング時に加える手法

研究の背景　拡散モデルの発展

- DDPM（2020）逆拡散過程でノイズを段階的
  - ADM（2021）クラス分類器を用いたガイド付き拡散
    - **Classifier-Free Guidance（2021）**
    - GLIDE（2021）CLIP を使ったテキスト条件付け

1：VAE

2：拡散モデル（Diffusion Model）

2.1：階層型VAE

2.2：順過程（Forward Process）

2.3：逆過程（Reverse Process）

3：DDPM / ADM の成立

4：分類器不要の条件付き拡散モデル

5：自然言語プロンプト統合

6：潜在空間拡散(Diffusion Modelへの発展)

Classifier-Free Guidance（2021）

ADMでは...

各ステップ t でのノイズのある画像 x_t からクラス予測分類器 $p_\emptyset(y \mid x_t, t)$ を学習する必要性,予測分類器勾配計算 $\nabla_{x_t} log\, p_\emptyset(y \mid x_t, t)$ でコストが増大

生成時にクラス分類器を使わないガイダンス手法を考案

ADMの各ステップにおける逆過程サンプリング

$$x_{t-1} = \mu_\theta(x_t, t) + s\,\beta_t \nabla_{x_t} \log p_\phi(y \mid x_t, t) + \sqrt{\beta_t}\, z, \quad z \sim \mathcal{N}(0, I).$$

条件 y 付き推定の導入　　無条件 $\emptyset$ 推定の導入

$\hat{\varepsilon}_{cond} = \varepsilon_\theta(x_t, t; y)$　　$\hat{\varepsilon}_{uncond} = \varepsilon_\theta(x_t, t; \emptyset)$

Classifier-Free Guidanceの平均μ_θと各ステップにおける逆過程サンプリング

$\hat{\varepsilon}_{guided} = \hat{\varepsilon}_{cond} + w(\hat{\varepsilon}_{cond} - \hat{\varepsilon}_{uncond})$　　w で条件付き/無条件 推定の比率を制御

$$\mu_{\text{guided}}(x_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\,\hat{\epsilon}_{\text{guided}}\right).$$

$$x_{t-1} = \mu_{\text{guided}}(x_t, t) + \sqrt{\beta_t}\, z, \quad z \sim \mathcal{N}(0, I).$$

ADMで外部化していたクラスガイドを内部化

研究の背景　拡散モデルの発展

- DDPM（2020）逆拡散過程でノイズを段階的
  - ADM（2021）クラス分類器を用いたガイド付き拡散
    - Classifier-Free Guidance（2021）
    - GLIDE（2021）CLIP を使ったテキスト条件付け

1：VAE

2：拡散モデル（Diffusion Model）

2.1：階層型VAE

2.2：順過程（Forward Process）

2.3：逆過程（Reverse Process）

3：DDPM / ADM の成立

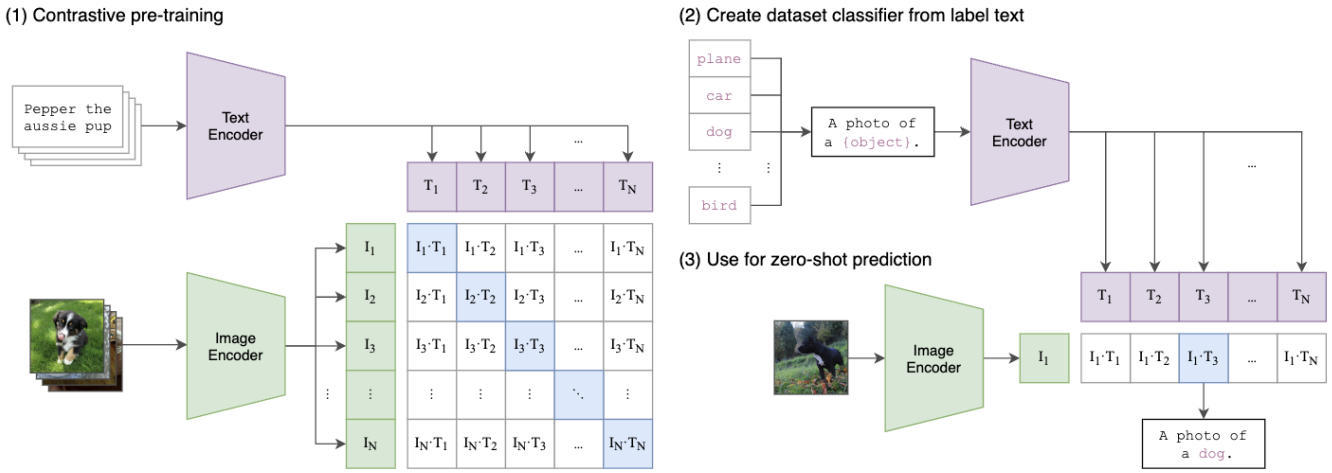
4：分類器不要の条件付き拡散モデル

5：自然言語プロンプト統合

6：潜在空間拡散(Diffusion Modelへの発展)

GLIDE（2022）

CLIP エンコーダで自然言語プロンプトをテキスト埋め込みに変換



入力テキストと**入力画像**で**CLIP**を事前学習　埋め込み画像との**コサイン類似度**で**CLIP**を決定

U-Net（拡散モデル本体）にテキスト埋め込み $\tau CLIP(y)$ を挿入

クリップ $\tau CLIP(y)$ 付き推定の導入

$$\hat{\varepsilon}_{cond} = \varepsilon_\theta(x_t, t; \tau CLIP(y))$$

無条件 $\emptyset$ 推定の導入

$$\hat{\varepsilon}_{uncond} = \varepsilon_\theta(x_t, t; \emptyset)$$

GLIDEの各ステップにおける逆過程サンプリング

$$x_{t-1} = \mu_{\text{guided}}(x_t, t) + \sqrt{\beta_t}\, z, \qquad z \sim \mathcal{N}(0, I).$$

研究の背景　拡散モデルの発展

- Classifier-Free Guidance (2021)
- GLIDE (2021) CLIP を使ったテキスト条件付け
  - Stable Diffusion (2022) [潜在変数系拡散モデル]

潜在空間拡散(Latent Diffusion)の導入

1 : VAE

2 : 拡散モデル（Diffusion Model）

2.1 : 階層型VAE

2.2 : 順過程（Forward Process）

2.3 : 逆過程（Reverse Process）

3 : DDPM / ADM の成立

4 : 分類器不要の条件付き拡散モデル

5 : 自然言語プロンプト統合

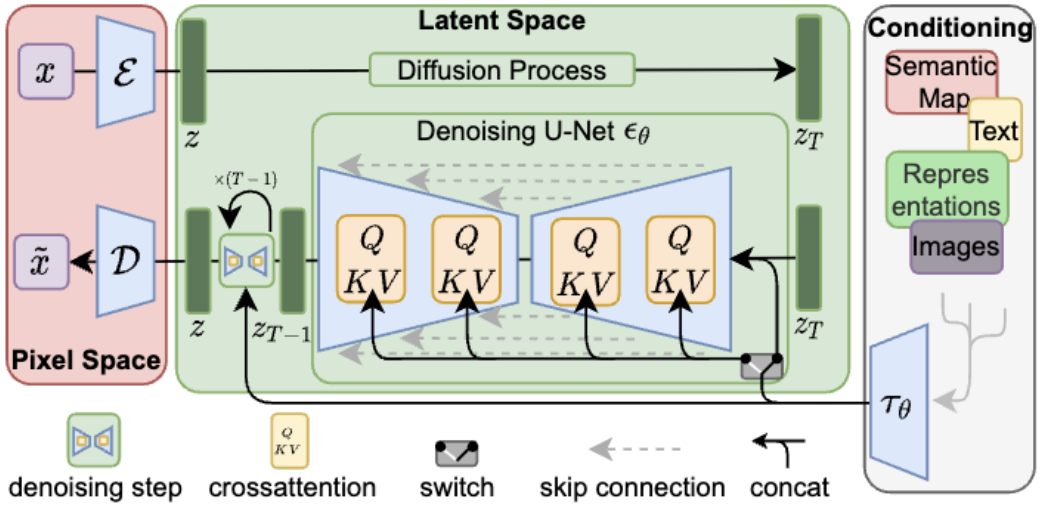
6 : 潜在空間拡散(Diffusion Modelへの発展)

Stable Diffusion (2022)

GLIDEでは...

512×512 以上の解像度を扱うには**数百億の FLOPs, 数百 GB のメモリ**が必要

Latent Diffusion



入力ピクセルをVAEで潜在変数化することで計算量を削減

GLIDEまでは

　画像ピクセルに**ガウスノイズ**を乗っけていた

Stable Diffusionでは

　潜在空間内で(潜在)変数を**拡散過程** / CLIPガイドに沿って**逆過程**

研究の発展 Face Personalize 拡散モデルの発展

Face Personalize 拡散モデルのこれまで

年代	著者	タイトル	アプローチ	備考
2022	Gal et al.	Textual Inversion	1単語の学習で顔表現	微調整が必要【17】
2022	Ruiz et al.	DreamBooth	個人単位の全体微調整	時間・計算コスト大【46】
2023	Kumari et al.	CustomDiffusion	Cross-attention層の軽量学習	学習は速いが過学習しやすい【30】
2023	Tewel et al.	Perfusion	super-classを用いた制御	過学習を緩和【54】
2023	Gal et al.	**E4T**	事前学習済エンコーダ＋軽微調整	速く編集性も高い【18】
2023	Yuan et al.	**Celeb Basis**	有名人をベースに顔を埋め込み	identity表現に特化【63】
2023	Xiao et al.	FastComposer	局所注意により多人数混在を防止	identity fidelity 低め【60】
2023	Wei et al.	ELITE	大規模事前学習でチューニング不要	高効率・汎用性重視【59】
2023	Chen et al.	**Dream Identity**	多スケール特徴＋自己学習	テスト時チューニング不要【11】
2024	Shiohara & Yamasaki	**Face2 Diffusion (F2D)**	MSID + Expression Guidance + CGDR	identityとtextの両立を実現（本論文）

GAN (2017–2019)

---------- ↓拡散モデルの始まり ----------

- DDPM (2020)
  - ADM (2021)
    - Classifier-Free Guidance (2021)
    - GLIDE (2021)
      - DALL·E 2 (2022)
        - DALL·E 3 (2023)
      - Imagen (2022)
        - Imagen 2 (2023)
          - Imagen 3 (2024)

---------- ↓個人特徴量系拡散モデル ----------

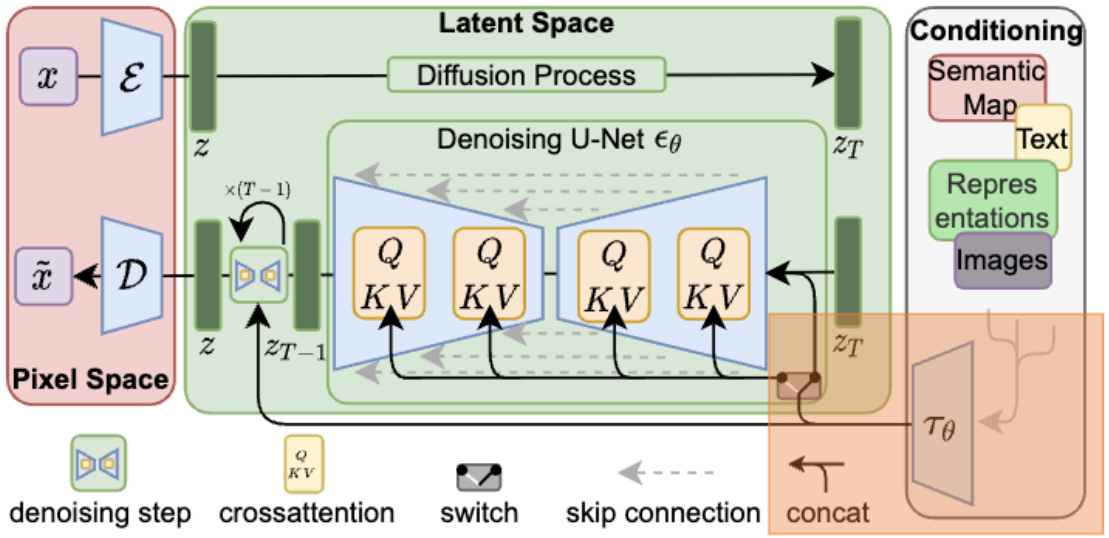
      - Stable Diffusion (2022)
        - 直接最適化系 (Direct Optimization)
          - Textual Inversion (2022)
          - DreamBooth (2022)
          - CustomDiffusion (2023)
          - Perfusion (2023)
        - エンコーダベース最適化系 (Encoder-Based Optimization)
          - E4T (2023)
          - CelebBasis (2023)
          - DreamIdentity (2023)
          - Face2Diffusion (2024)

研究の背景 Face Personalize 拡散モデルの発展

1：直接最適化系
(Direct Optimization)

2：エンコーダベース最適化系
(Encoder-Based Optimization)

3：最適化不要／高速化系
(Optimization-Free／Fast Tuning)



Input テキスト・画像処理部分での最適化を実装

Textual Inversion (2022)

擬似トークン埋め込みベクトルで
学習可能な **word embedding を最適化** 入力画像を再構築する

DreamBooth (2022)

Text to Image (T2I)モデル全体を少数ショットで
ファインチューニング　　新概念(**subject**)を挿入

CustomDiffusion (2023)

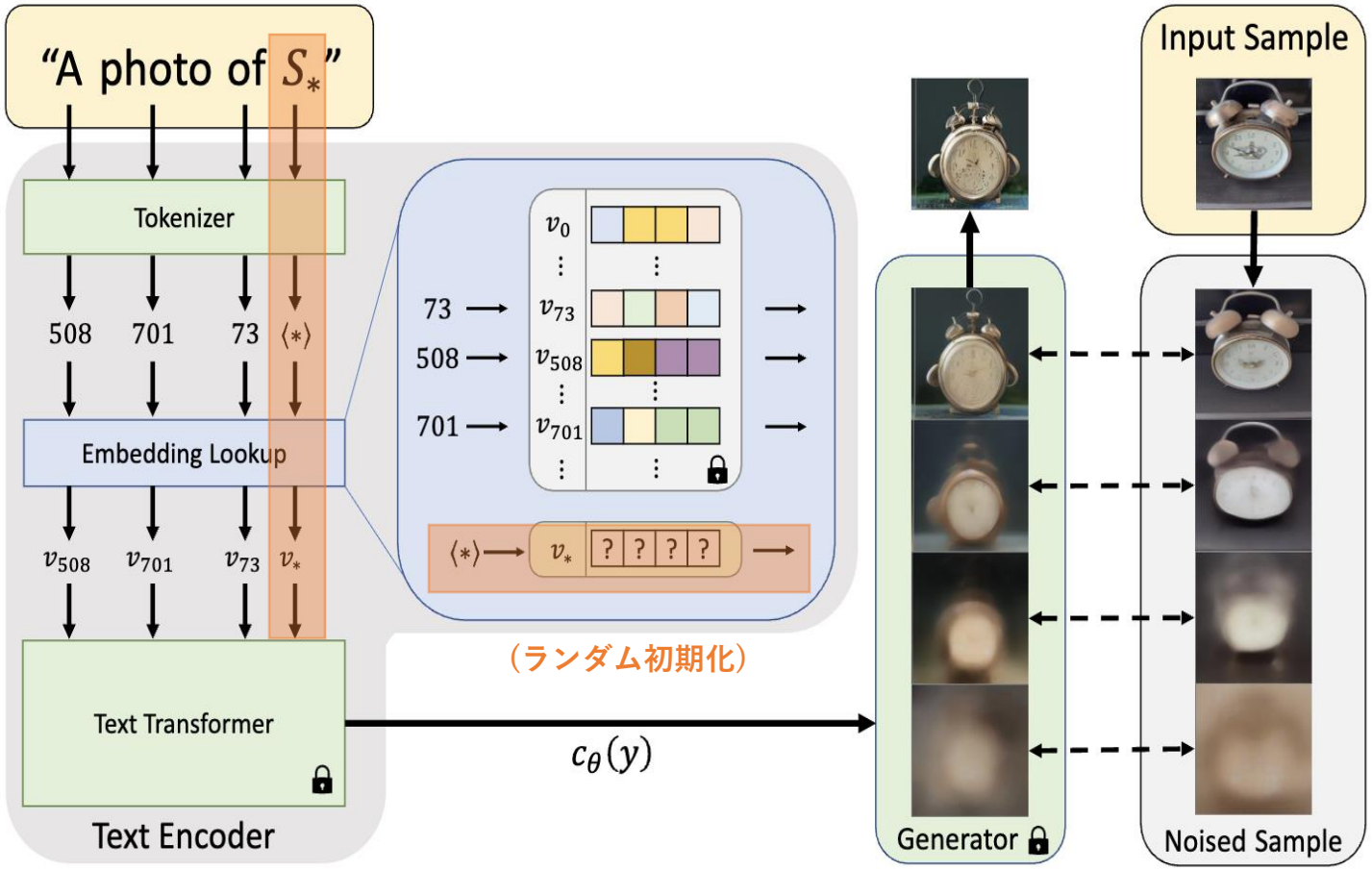
U-Net の cross-attention 層のみを軽量に
ファインチューニング　新概念を反映させる手法

Perfusion (2023)

cross-attention に **“super-class (a person)” 正則化**を導入して
過学習を緩和する手法

研究の背景

Face Personalize 拡散モデルの発展



Textual Inversion（2022）

擬似トークン埋め込みベクトル v_*

Step1

Embedding Lookup(埋め込み辞書)に 〈＊〉
を新トークンベクトルとして追加

Step2

〈＊〉以外のベクトルの重みは事前学習
から**更新しない**

Step3

更新したテキストから学習した**ガイダンス**
$CLIP(c_\theta(y))$ に沿って**"Input Sample"**を参照し
ながらデノイズしていく

研究の背景

Face Personalize 拡散モデルの発展

“A stock photo of a doctor” (Base model)

“A photo of S_*” (Ours)

(顔怖すぎ笑)

研究の背景

Face Personalize 拡散モデルの発展

1：直接最適化系

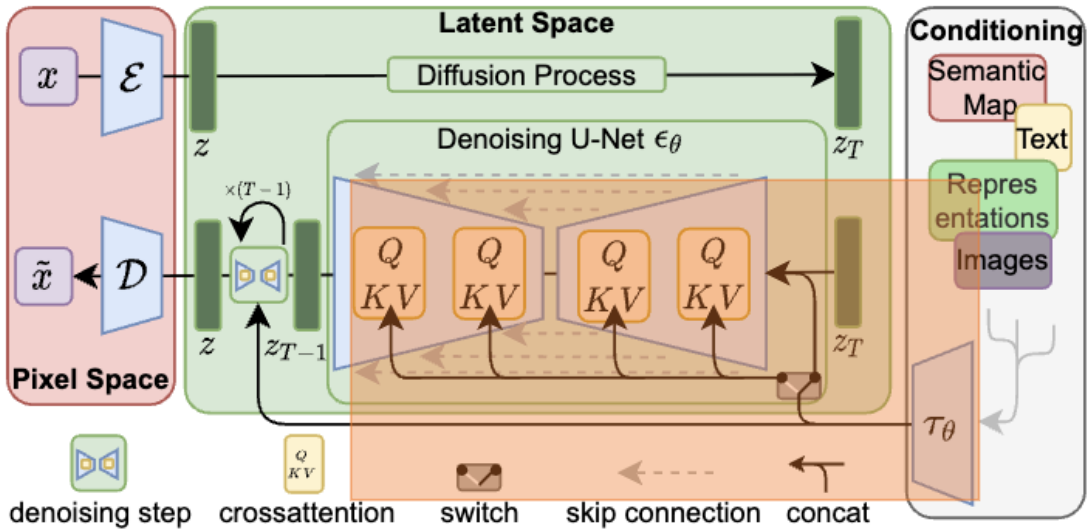
(Direct Optimization)

2：エンコーダベース最適化系

(Encoder-Based Optimization)

3：最適化不要／高速化系

(Optimization-Free／Fast Tuning)



U-Net 内のCross Attention 部分での最適化を実装

E4T (2023)

Stable Diffusion 本体は完全凍結． 外部 ID Encoder から得た顔特徴をマッピングしCross-Attention に渡す回路だけを最適化する

Celeb Basis (2023)

“テキスト埋め込み + Celeb Basis → ID 埋め込み” を結合して Cross-Attention に渡す

Dream Identity (2023)

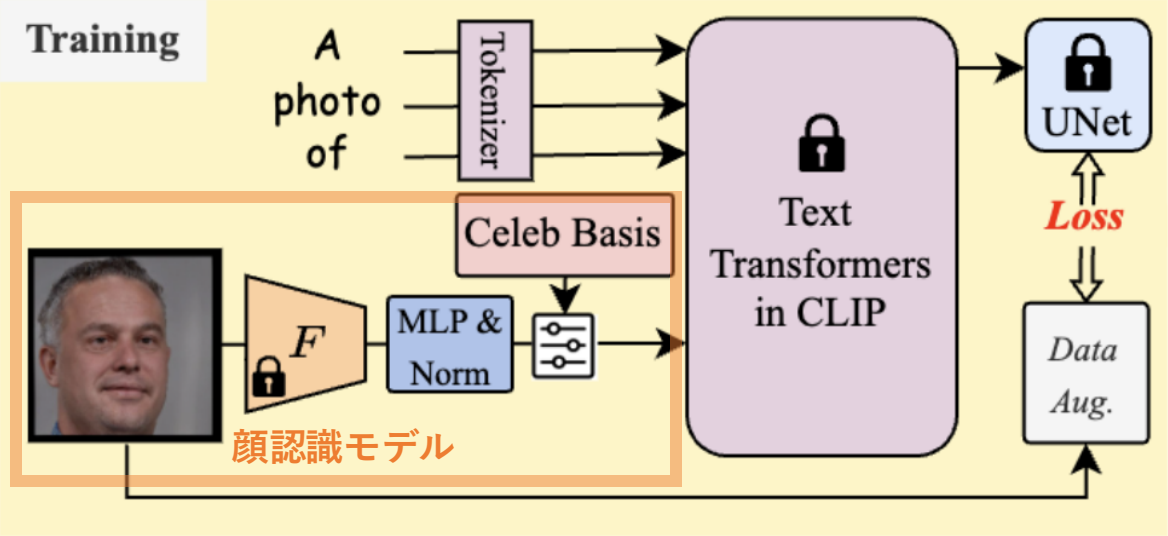
“テキスト埋め込み + Multi-Scale → ID 埋め込み” を結合して Cross-Attention の Key/Value に渡す

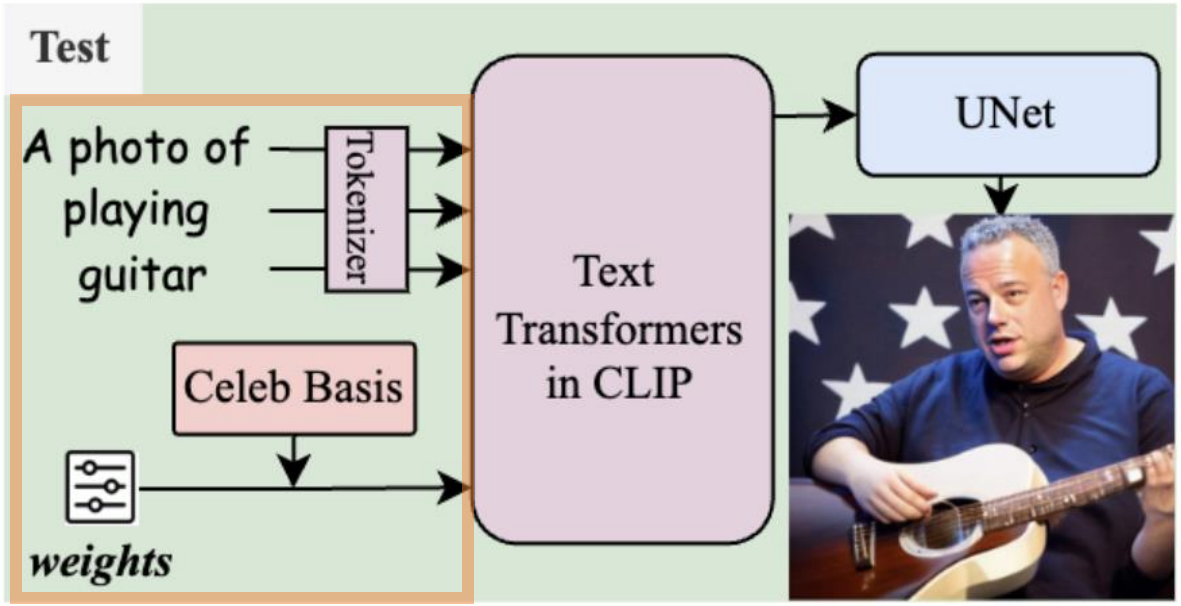
Face2Diffusion (2024)

“テキスト埋め込み + MSID → ID 埋め込み + Expression Guidance → 表情特徴 + CGDR 正則化” を結合して Cross-Attention に渡す

研究の背景

Face Personalize 拡散モデルの発展





Celeb Basis (2023)

“**ID 埋め込み** → テキスト埋め込み + Celeb Basis”
を結合して Cross-Attention に渡す

Step1

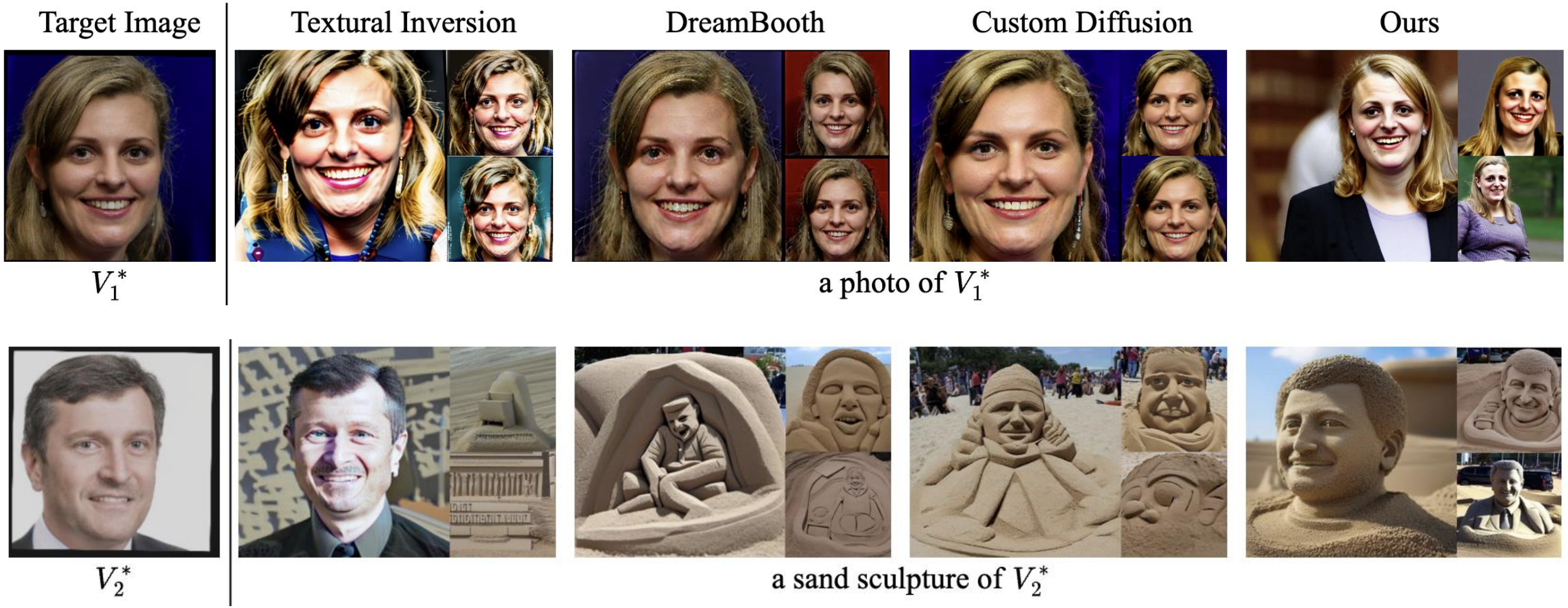
ID 埋め込みをマッピングネットワーク(**MLP & Norm**)
で SD 用トークン埋め込みの形に変換

Step2

Trainingで保存した**Celeb Basis Weights**と
入力プロンプトのCross Attentionでガイドを作成

研究の背景

Face Personalize 拡散モデルの発展



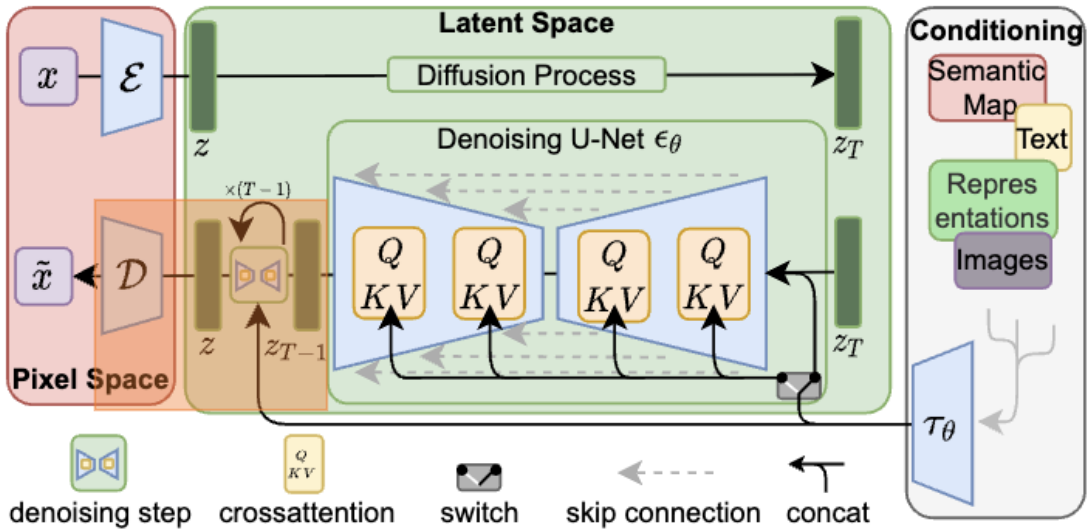
研究の背景

Face Personalize 拡散モデルの発展

1：直接最適化系
(Direct Optimization)

2：エンコーダベース最適化系
(Encoder-Based Optimization)

3：最適化不要／高速化系
(Optimization-Free／Fast Tuning)



Inference 時にのみ参照する最低限のモデルを構築

FastComposer (2023)

顔検出モジュール
“どこに顔があるか” を推定

局所 Cross-Attention Unit
検出した顔の RoI(Region of Interest) 内だけを重点的に Attention

ID embedding
ID embedding を「プロンプトに入れた状態で」U-Net に条件付け

ELITE (2023)

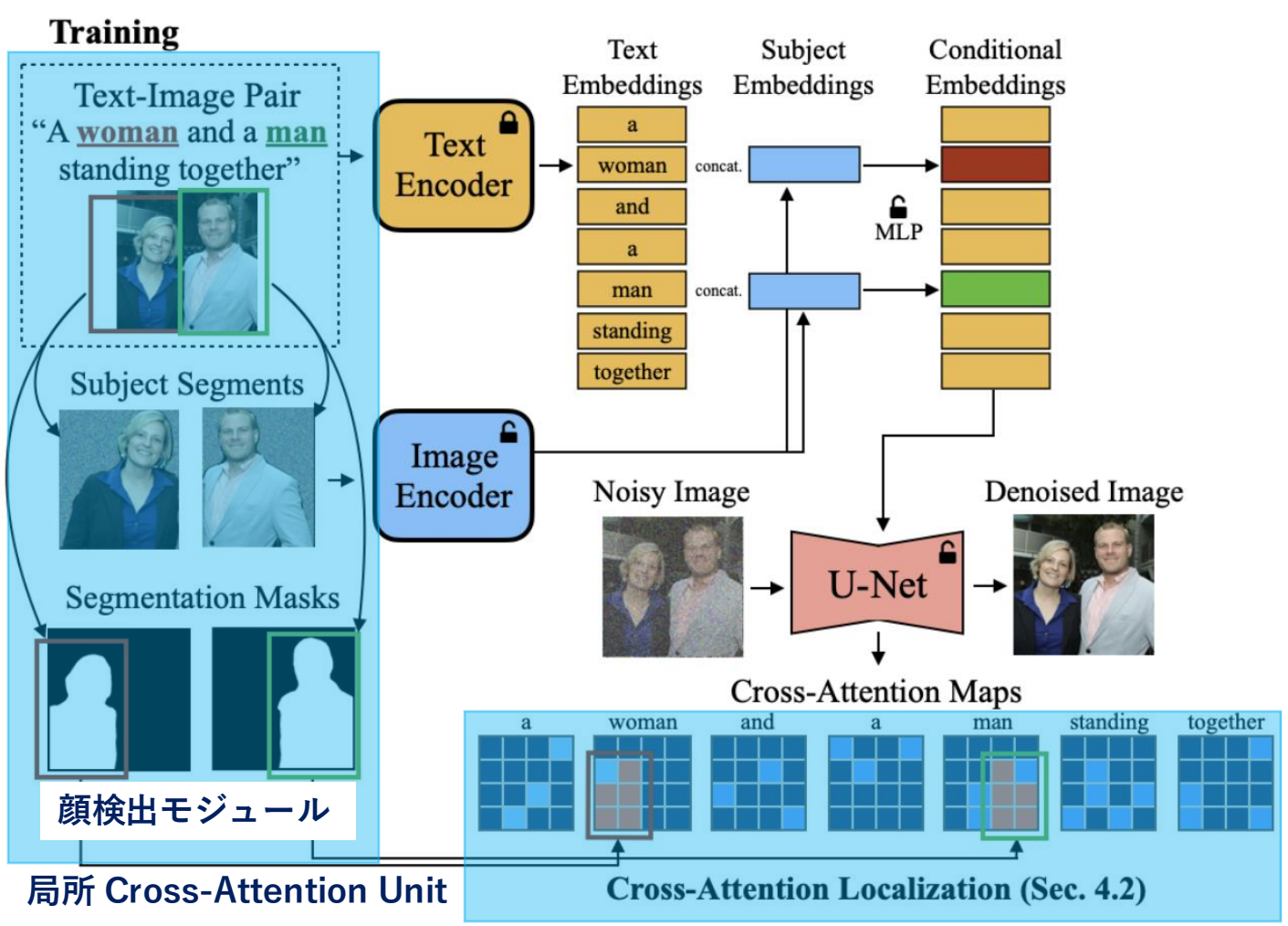
事前に
「数万～数十万枚の顔画像」と
「テキストプロンプト or CLIP テキスト埋め込み」
の対応データセットで訓練済み

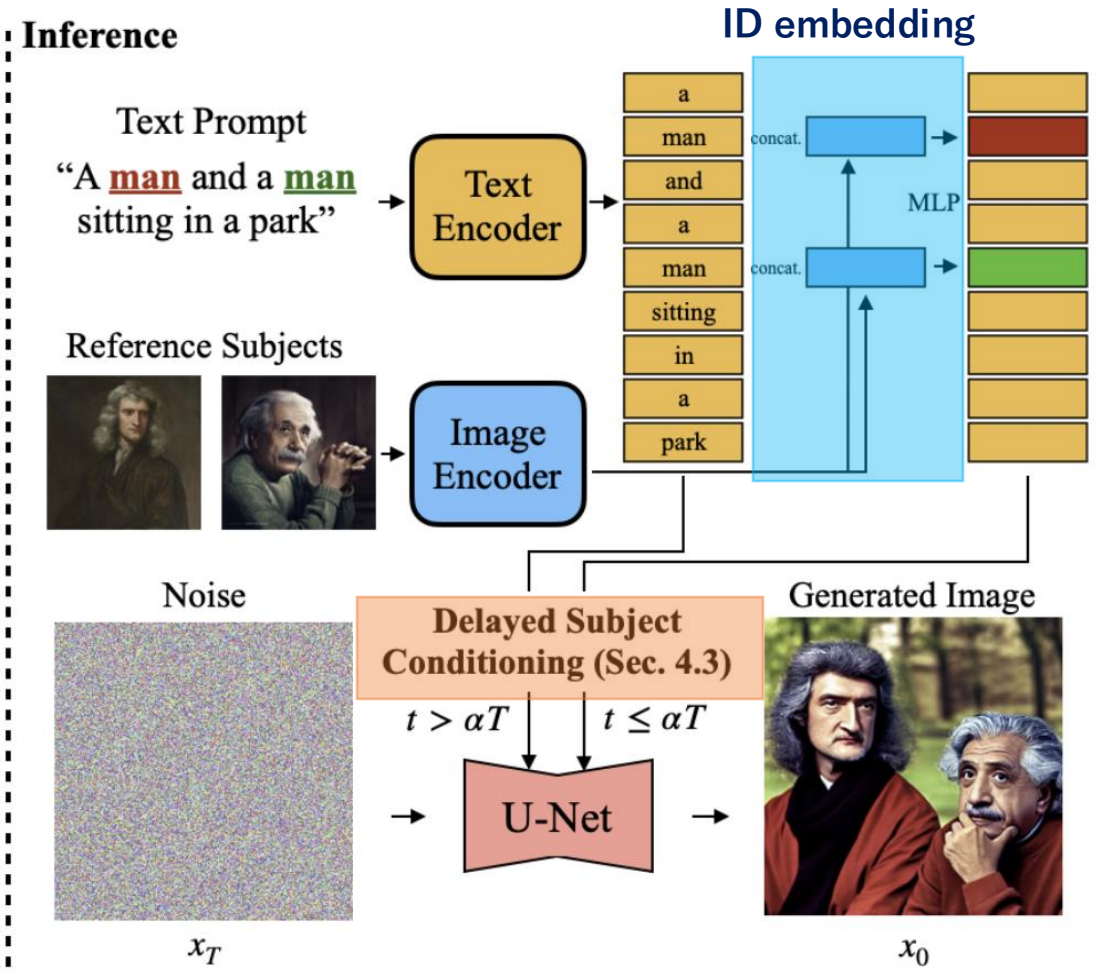
「本人顔画像 → CLIP ID 埋め込み → マッピングネットワーク」

研究の背景

Face Personalize 拡散モデルの発展

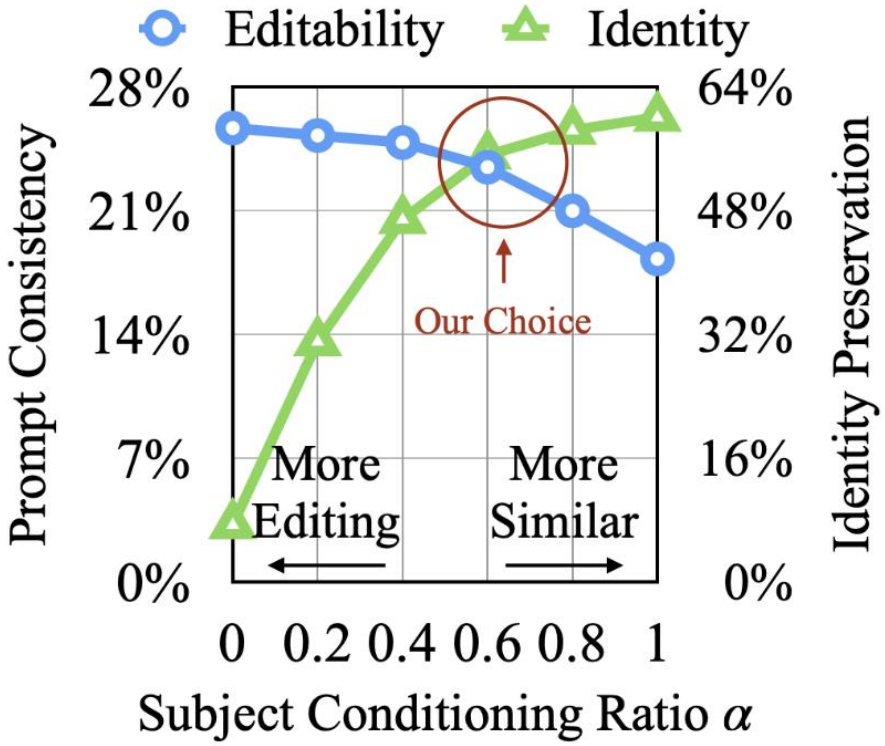
FastComposer (2023)

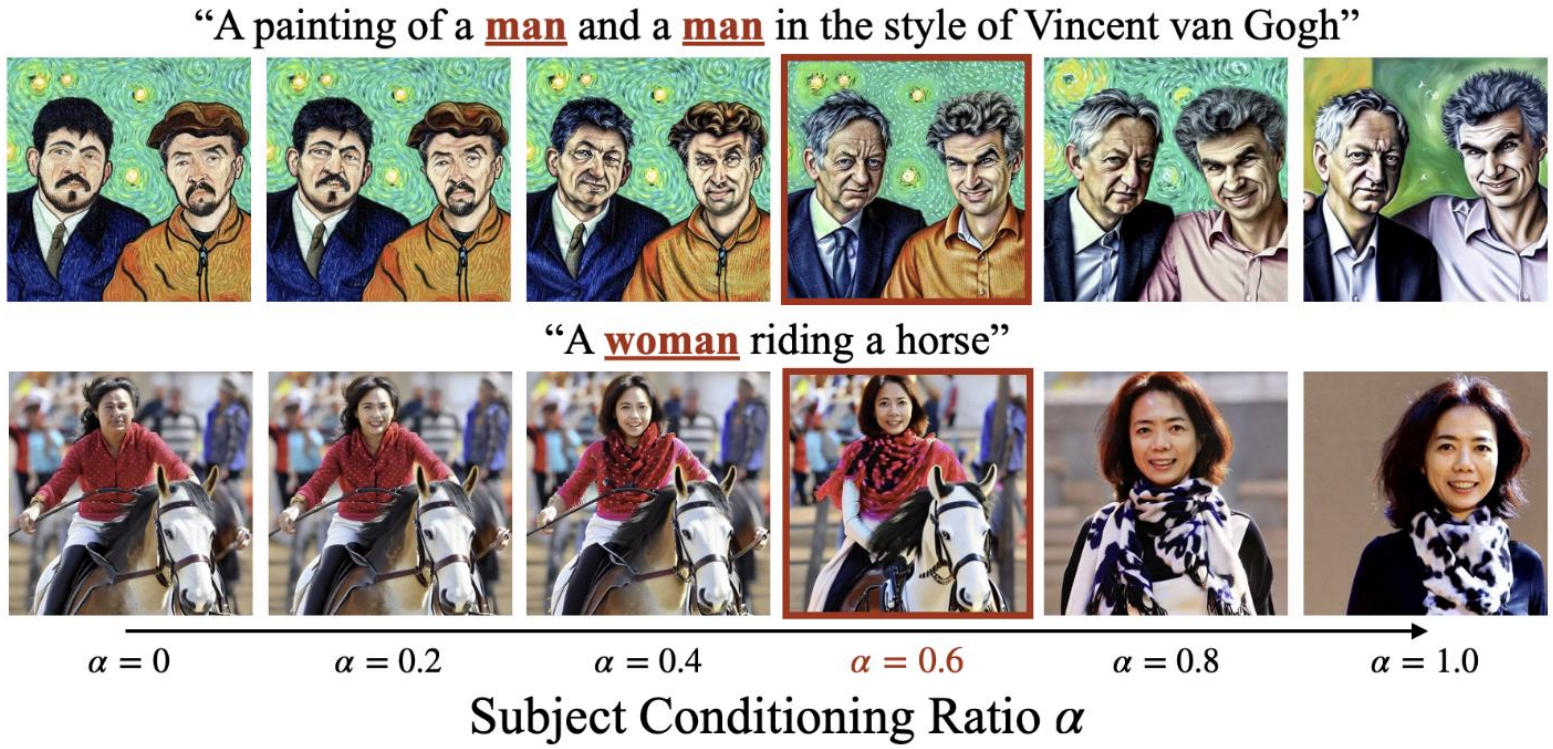




研究の背景

Face Personalize 拡散モデルの発展





3 : Methodology

3.1 : Multi-Scale Identity Encoder (MSID)

3.2 : Expression Guidance (EG)

3.3 : Class-Guided Denoising Regularization (CGDR)

研究の背景

Face Personalize 拡散モデルの発展

1：直接最適化系

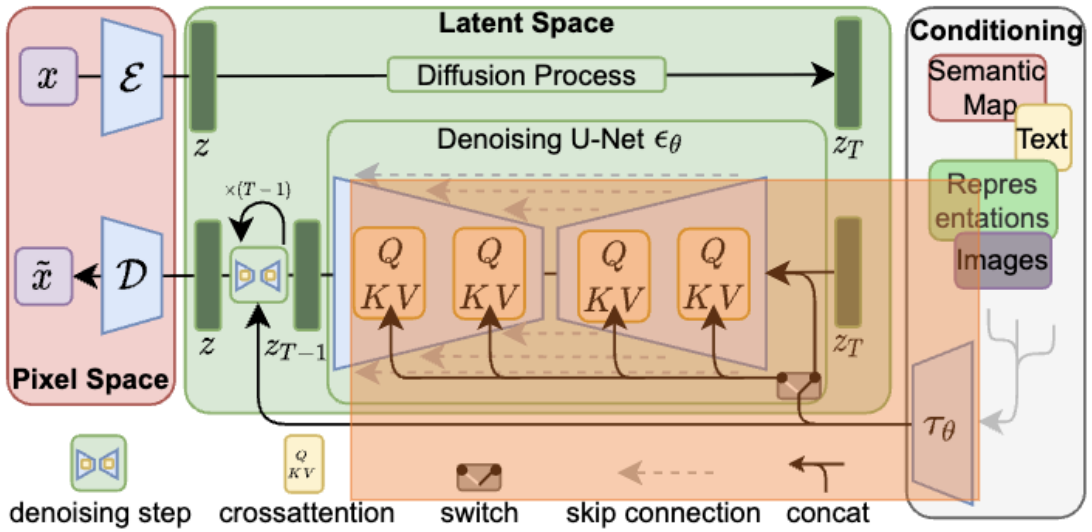
(Direct Optimization)

2：エンコーダベース最適化系

(Encoder-Based Optimization)

3：最適化不要／高速化系

(Optimization-Free／Fast Tuning)



U-Net 内のCross Attention 部分での最適化を実装

E4T (2023)

Stable Diffusion 本体は完全凍結． 外部 ID Encoder から得た顔特徴をマッピングしCross-Attention に渡す回路だけを最適化する

Celeb Basis (2023)

“テキスト埋め込み + Celeb Basis → ID 埋め込み” を結合して Cross-Attention に渡す

Dream Identity (2023)

“テキスト埋め込み + Multi-Scale → ID 埋め込み” を結合して Cross-Attention の Key/Value に渡す

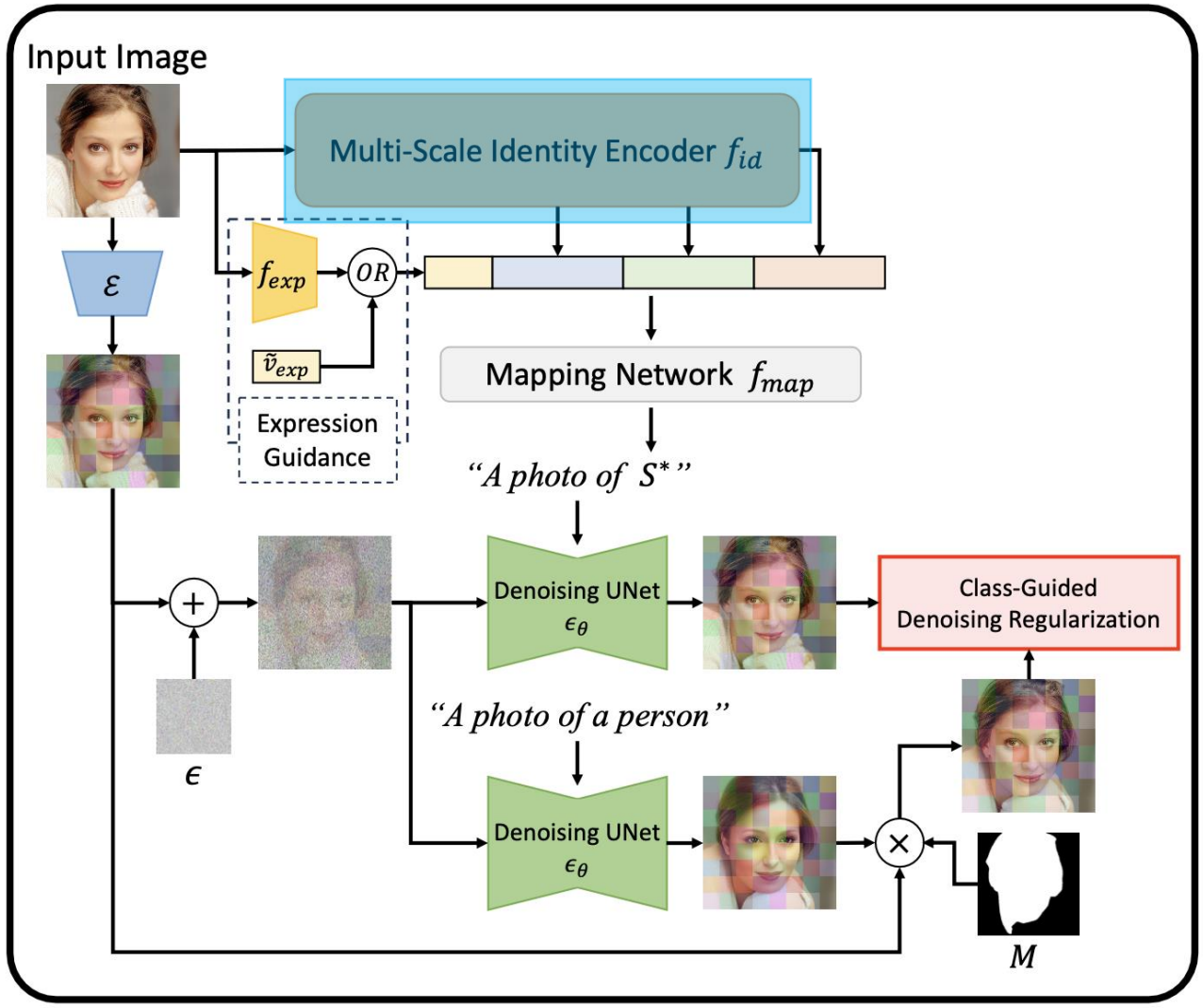
Face2Diffusion (2024)

“テキスト埋め込み + MSID → ID 埋め込み + Expression Guidance → 表情特徴 + CGDR 正則化” を結合して Cross-Attention に渡す

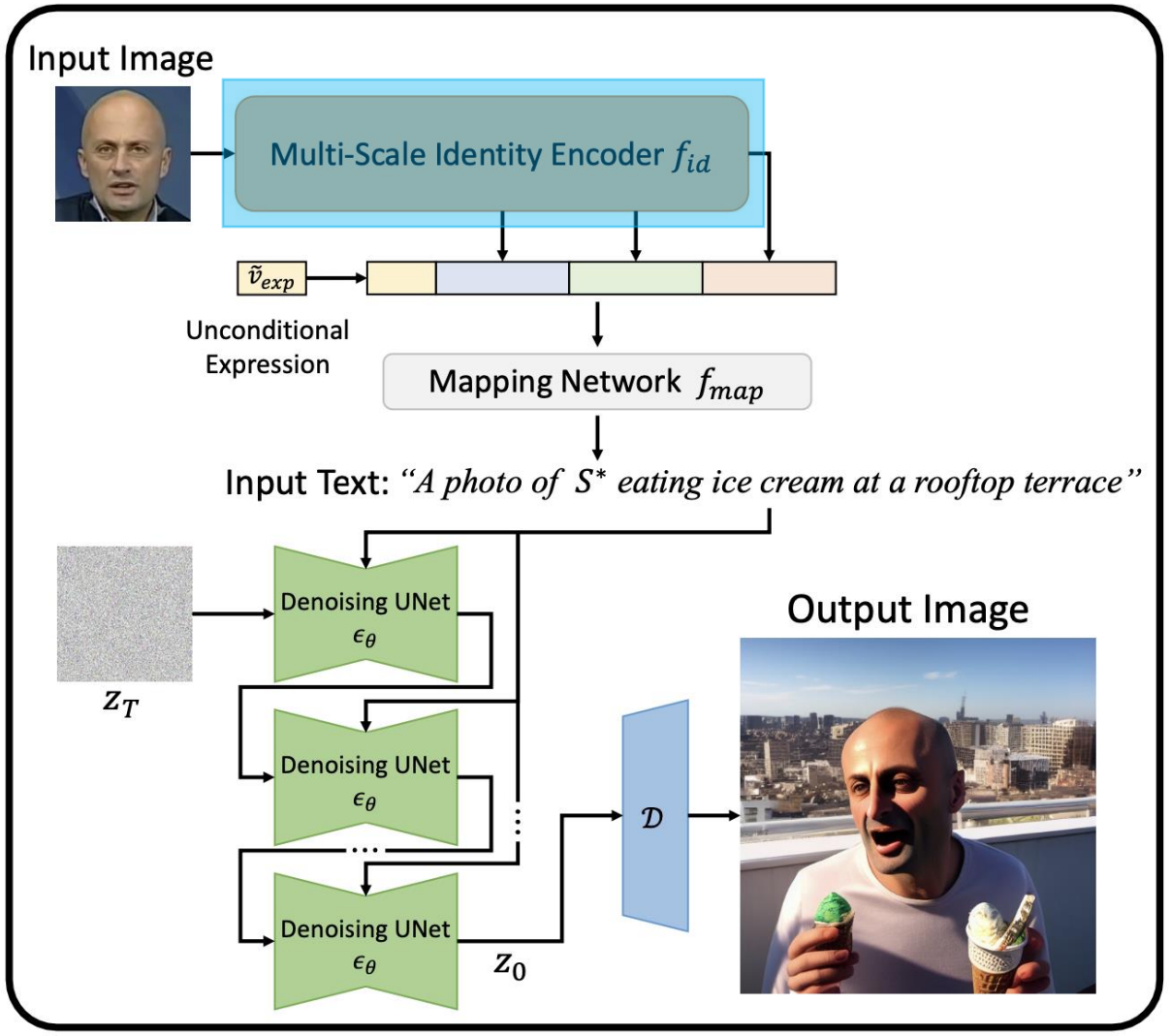
3.1： Multi-Scale Identity Encoder (MSID)

Architecture

Face 2 Diffusion



(a) Training



(b) Inference

Methodology

Face Personalize 拡散モデルの発展

Multi-Scale Identity Encoder (MSID)

Celeb Basis Model (2023)

顔認識モデルの有用性が示した

Dream Identity (2023)

浅い層から深い層までの**マルチスケール特徴**によって個人性の類似度が向上した

$$V = [v_3, v_6, v_9, v_{12}, v_{final}].$$

マルチスケール特徴ベクトル化

$$\mathcal{L}_{reg} = \sum_{i=1}^{k} \|s_i\|.$$

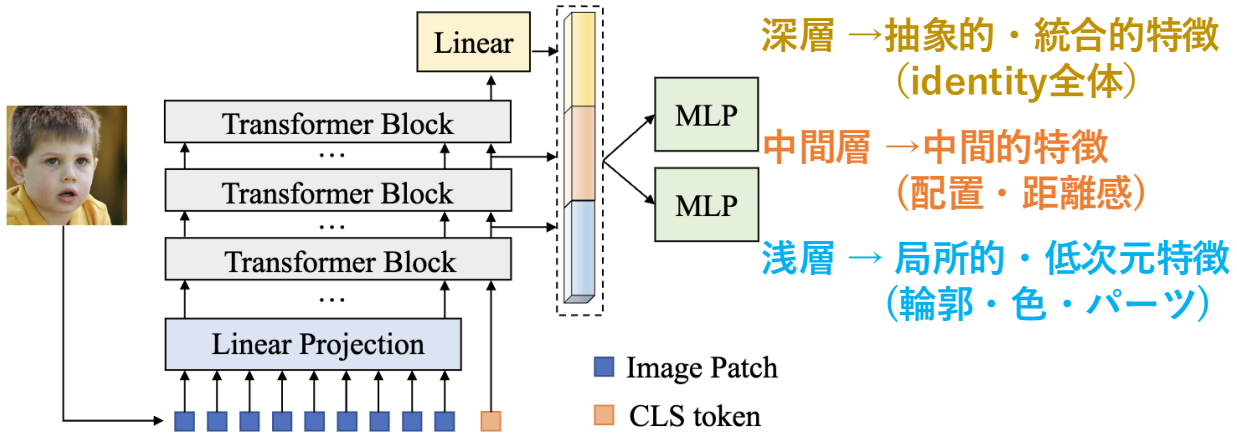
$$s_i = MLP_i(V), \text{for } i = 1, ..., k,$$

MLP（線形層）に通す

$$\mathcal{L}_{total} = \mathcal{L}_{diffusion} + \lambda \mathcal{L}_{reg},$$

ノイズ予測損失　正則化項

(c) M² ID Encoder



深層 →抽象的・統合的特徴（identity全体）

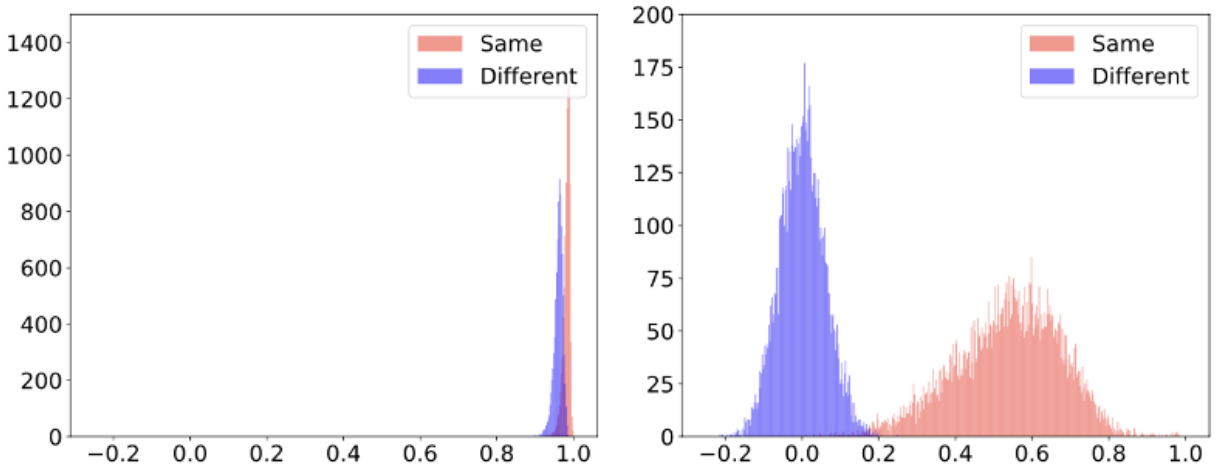
中間層 →中間的特徴（配置・距離感）

浅層 → 局所的・低次元特徴（輪郭・色・パーツ）

トレーニングデータセットの各個人について

{ **Anchor**, **Positive**(同一人物), **Negative**(異なる人物)}

のトリプレットを一様にサンプリング



(a) 3rd layer of ArcFace　(b) 12th layer of ArcFace

**横軸は個人性の類似度、縦軸は頻度

浅層. (3rd layer)：個人性の識別が困難(AUC = 93.99%)

深層(12th layer)：高い識別性能（AUC = 99.97%)

Methodology

Face Personalize 拡散モデルの発展

Multi-Scale Identity Encoder (MSID)

Dream Identity (2023)の定式化では浅層の識別が不十分
MSIDではその要因である**本質でない情報**
[カメラの向き(斜めか正面か), 表情(口が開いている, 笑っている), 照明や背景]
を除き判別の制度を向上させる

特徴 v_{id} と重み W_y を単位ベクトルとして正規化

$\cos(W_y^T v_{id})$ が内積＝角度の類似度　　$v_{id} = [v_{id}^1, v_{id}^2, ..., v_{id}^D]$ where v_{id}^i

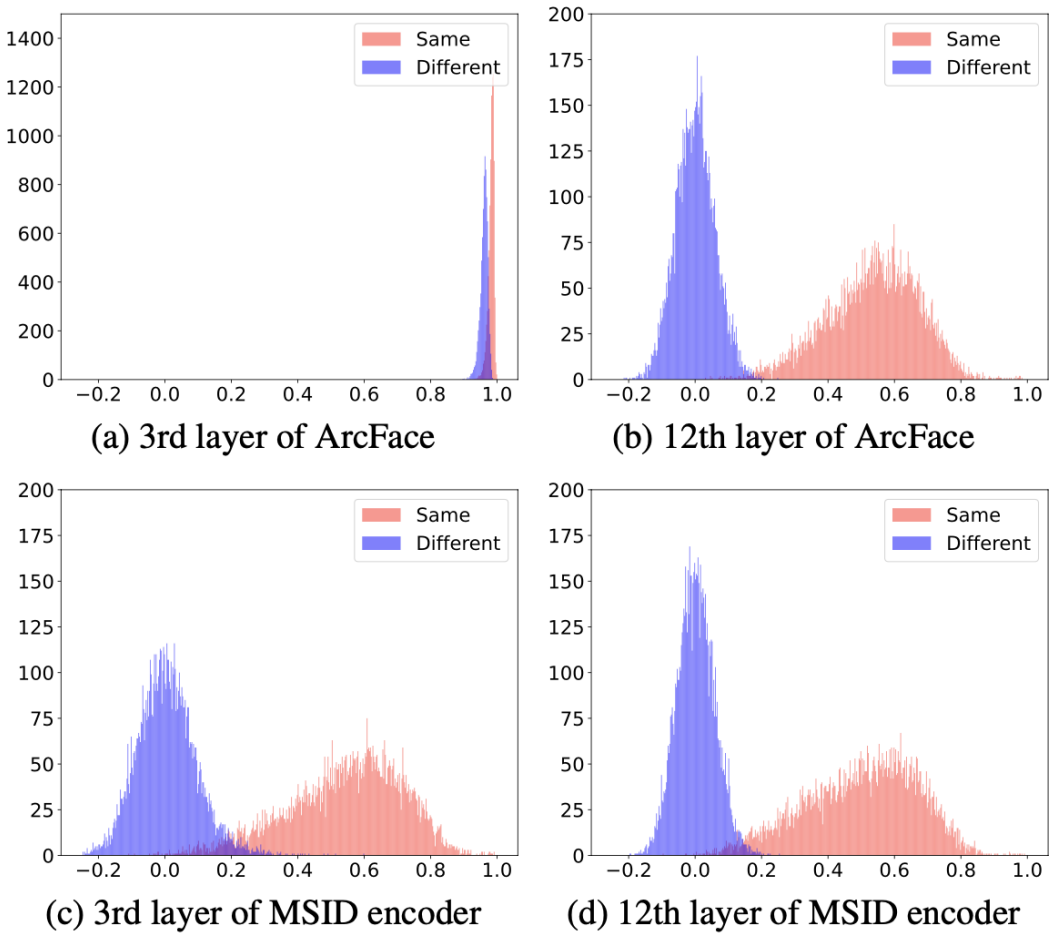
正解クラスにだけ角度的なマージンmを加える

$$\mathcal{L}_m = -\log \frac{e^{s\cos(W_y^T v_{id}+m)}}{e^{s\cos(W_y^T v_{id}+m)} + \sum_{k=1,k\neq y}^{K} e^{s\cos W_k^T v_{id}}}$$

正解クラスyのスコア

全クラスのスコアの合計

v_{id}	エンコーダで抽出された顔の特徴ベクトル（正規化済み）
W_y	クラス y の重みベクトル（正規化済み）
W_k	他のクラス k の重みベクトル（同様）
s	スケーリング係数（cos関数の鋭さ調整）
m	マージン　クラス yとの距離に罰則を与える



(a) 3rd layer of ArcFace　(b) 12th layer of ArcFace

(c) 3rd layer of MSID encoder　(d) 12th layer of MSID encoder

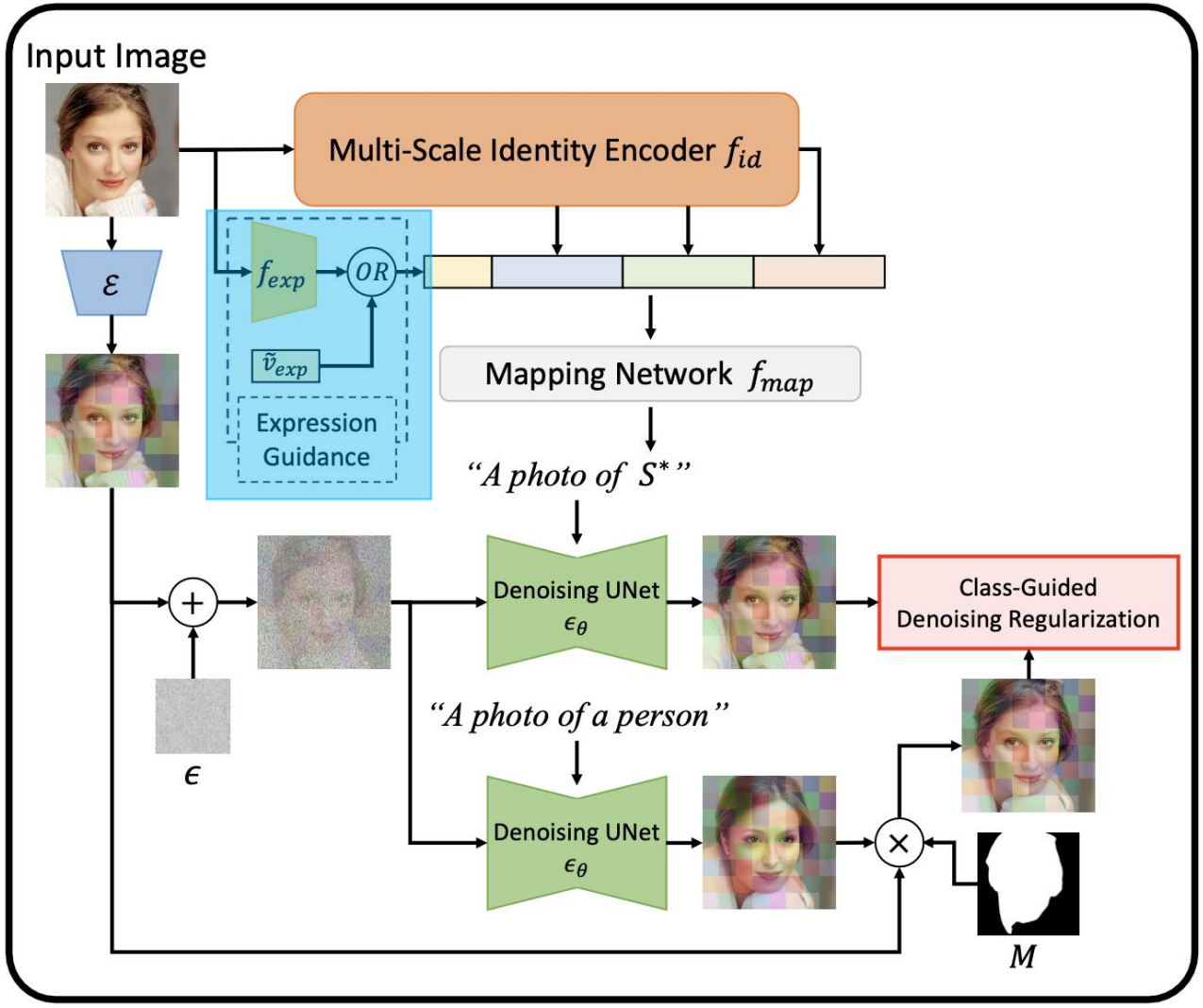
浅層. (3rd layer)：識別が大幅上昇(AUC = 99.46%)

深層(12th layer)：高い識別性能（AUC = 99.97%)

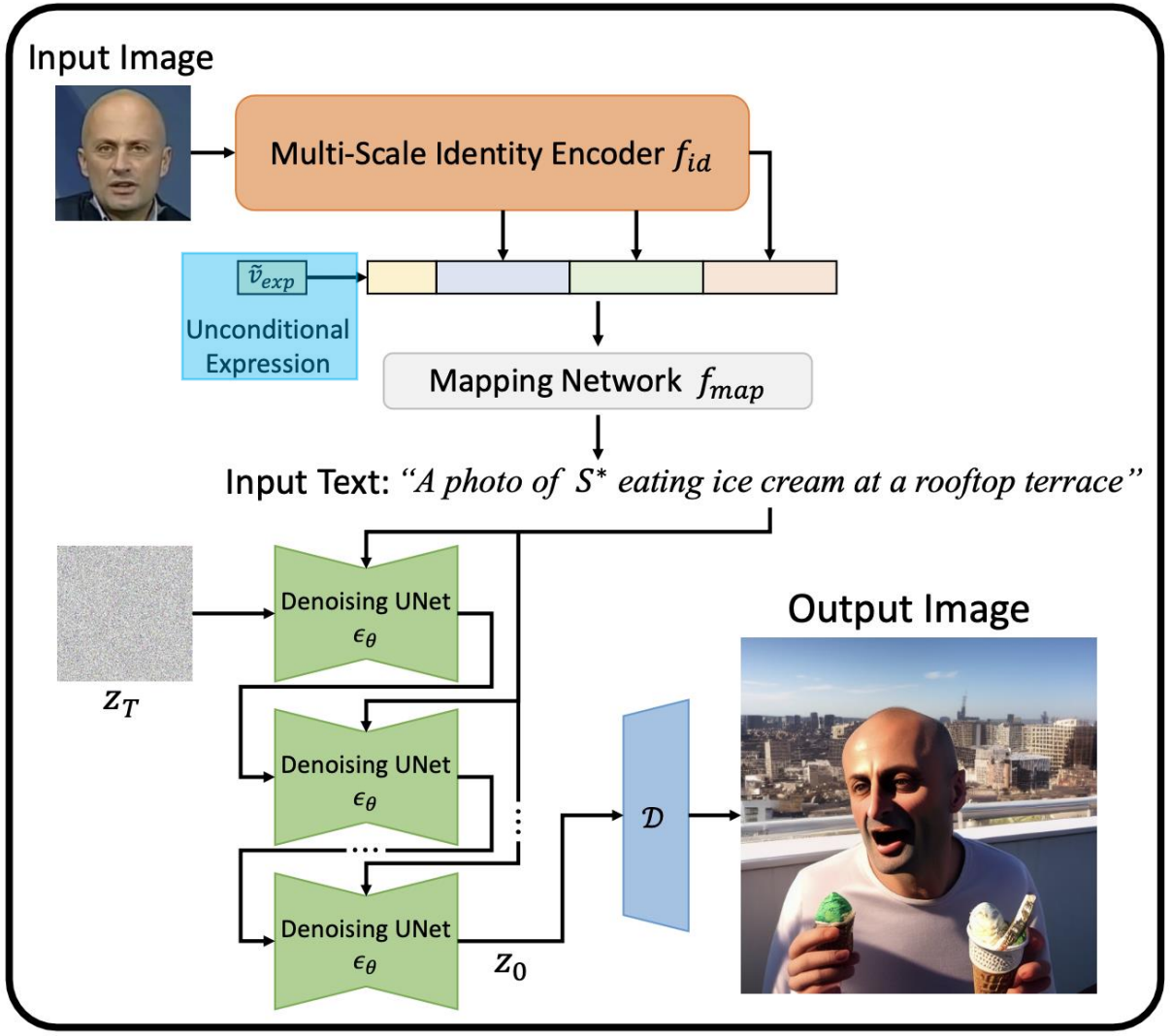
3.2： Expression Guidance (EG)

Architecture

Face 2 Diffusion



(a) Training



(b) Inference

Methodology

Face Personalize 拡散モデルの発展

Expression Guidance (EG)

顔の表情が入力画像のものに固定されてしまい

テキストに書かれた感情“S* looking shocked”

などが反映されにくい

3D再構成モデルによる
表情特徴 + Dropout
→ Mapping Network

$$v_{\text{exp}} = f_{\text{exp}}(x)$$

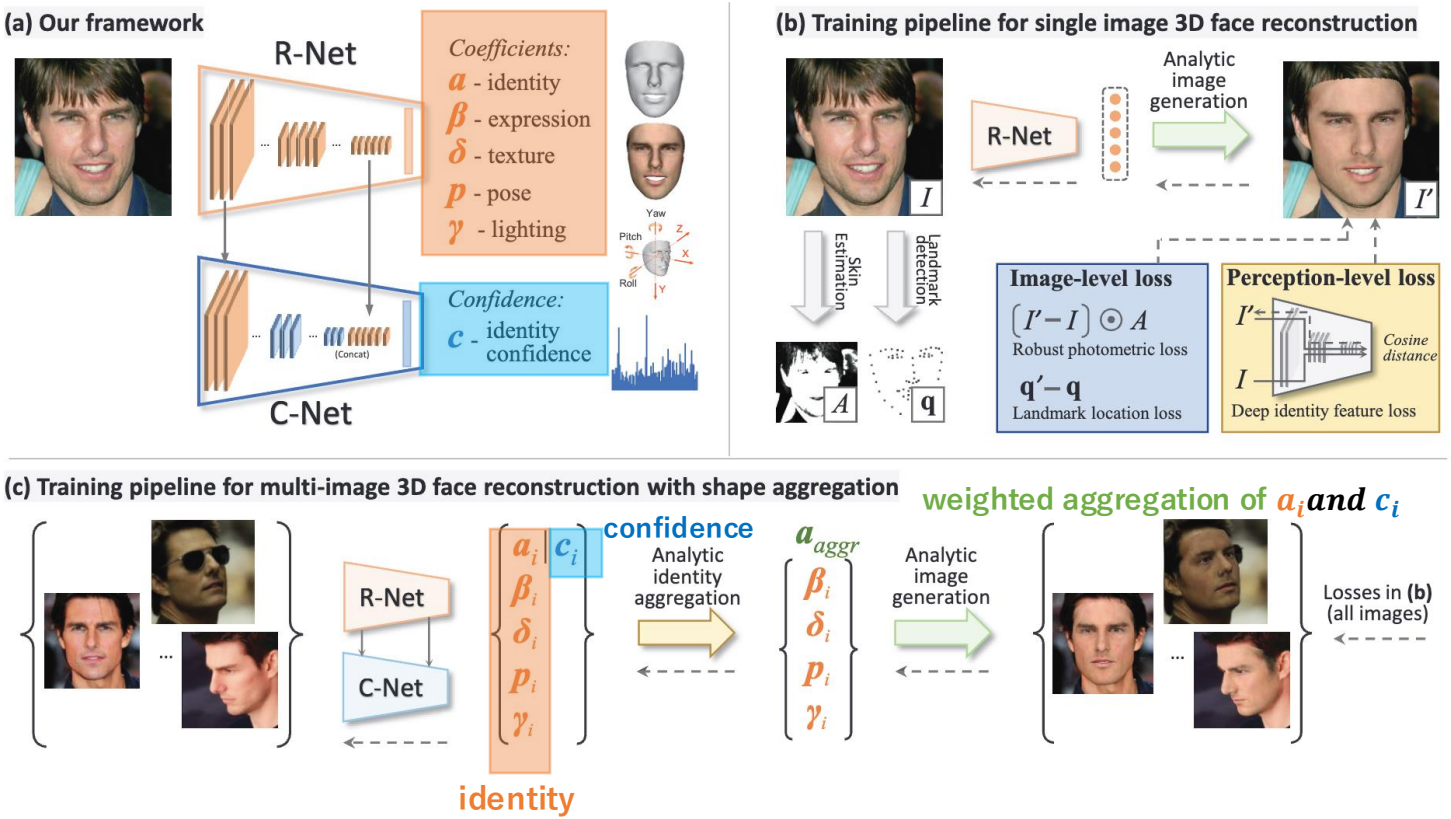
$$S^* = \begin{cases} f_{map}([v_{id}, v_{exp}]) & (p = 0.8), \\ f_{map}([v_{id}, \tilde{v}_{exp}]) & (p = 0.2), \end{cases}$$

学習時に 20% で実際の表情特徴 v_{exp}

の代わりに $\tilde{v}_{exp}$を使用

3D再構成モデル　2019年

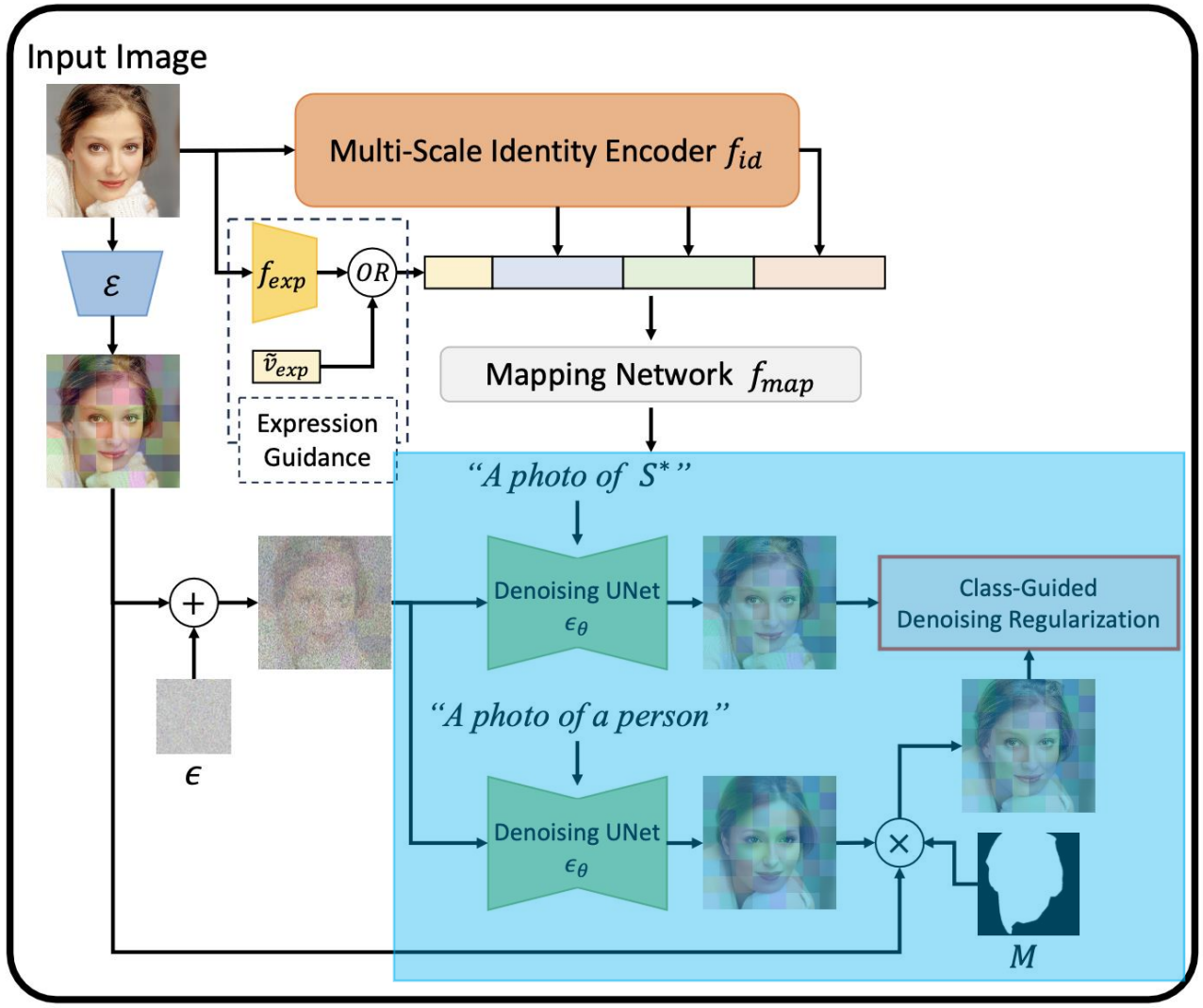
Accurate 3D Face Reconstruction with Weakly-Supervised Learning: From Single Image to Image Set



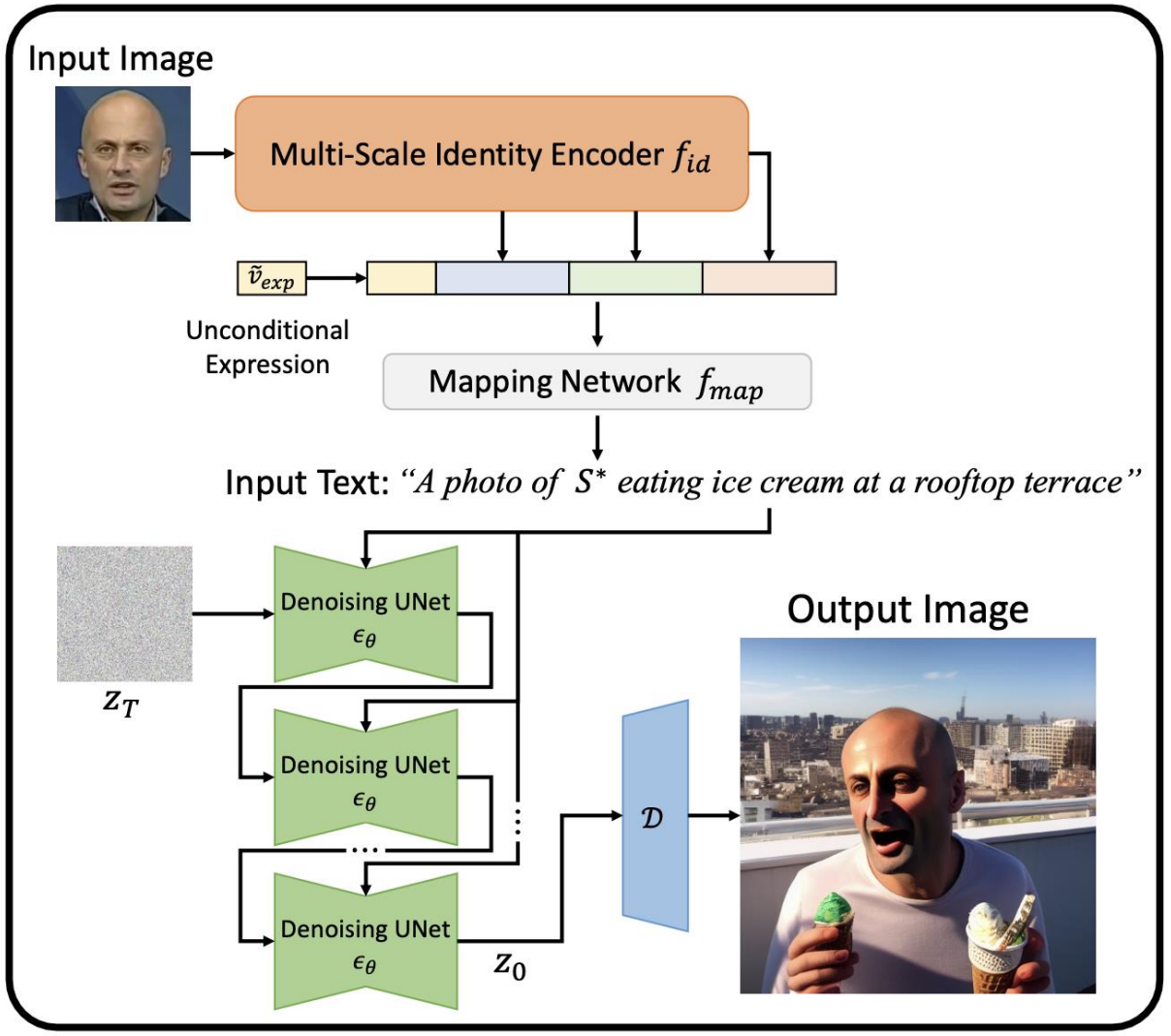
3.3：Class-Guided Denoising Regularization (CGDR)

Architecture

Face 2 Diffusion



(a) Training



(b) Inference

Methodology

Face Personalize 拡散モデルの発展

Class-Guided Denoising Regularization (CGDR)

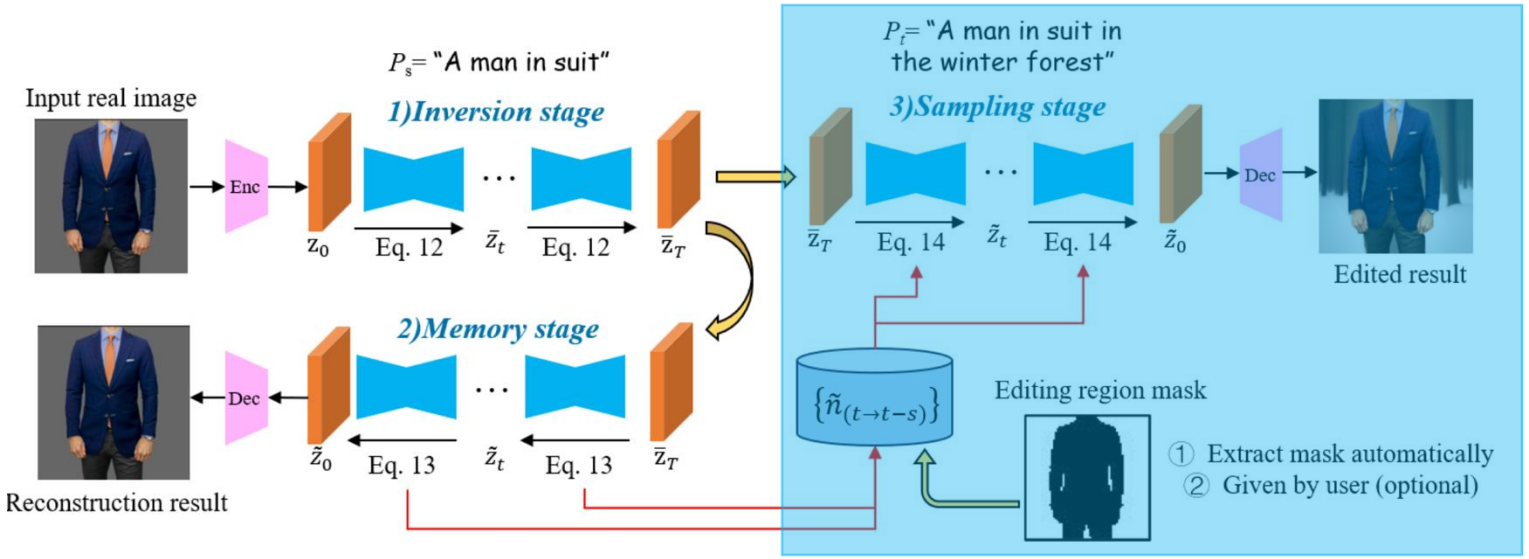
既往のClass – Guided Denoising は拡散過程において

生成した背景が**入力画像に過剰適合**する傾向にある

既存手法： Delayed Subject Conditioning

$$\hat{\epsilon}_t = \epsilon_\theta(z_t, t, c) \qquad c_t = \begin{cases} c_{\text{neutral}} & \text{if } t \le t_{\text{switch}} \\ c_{\text{subject}} & \text{if } t > t_{\text{switch}} \end{cases}$$

$c_{neutral}$	一般的なクラス語の埋め込み (“$a\ person$”)
$c_{subject}$	主題の埋め込み(特定の人物の特徴 “$S*$” など)
t_{switch}	主題の条件付けを開始する時間ステップ



Problems

どのタイムステップで背景を含む
テキストプロンプトを挿入するかを決定する必要性

顔を含む生成の場合 identity が崩壊する

Methodology

Face Personalize 拡散モデルの発展

Class-Guided Denoising Regularization (CGDR)

$$\hat{\epsilon} = \epsilon_\theta(z_t, t, \tau(p)), \quad \hat{\epsilon}_c = \epsilon_\theta(z_t, t, \tau(p_c))$$

特定人物の顔 identity に近づける (“of S* ”)

人か背景かの識別(“a person”)

$$\mathcal{L} = \|\hat{\epsilon} - (\epsilon \odot M + \hat{\epsilon}_c \odot (1 - M))\|_2^2 \qquad M = \begin{cases} 1：顔領域 \\ 0：背景領域 \end{cases}$$

$\tau(p)$	$c_{id} = f_{id}(x)$
$\tau(p_c)$	クラス語プロンプトのCLIP(”$a\ person$”)
ε_θ	拡散モデルの条件付きノイズ予測器
$\hat{\varepsilon}$	通常プロンプトに基づくノイズ予測
$\hat{\varepsilon_c}$	クラス語プロンプトに基づくノイズ予測
ε	ground truth ノイズ（訓練時に与えられる）
M	セグメンテーションマスク(顔領域:1, 背景:0)

$M = 1$ ：顔領域

$$\mathcal{L} = \|\hat{\varepsilon} - \varepsilon\|_2^2$$

identity（顔）は ground truth ノイズに基づいて復元

identity fidelity を保つための通常の拡散損失

$M = 0$ ：背景領域

$$\mathcal{L} = \|\hat{\varepsilon} - \hat{\varepsilon_c}\|_2^2$$

背景部分は「a person」プロンプトによる生成挙動に近づける

背景に「入力画像の過学習」が起こるのを防ぐ

損失関数が最小化されたε_θを乗せたガウスノイズを乗っける >>> DDPMと一緒

4 : Experiments

比較結果

過去モデルとの定量比較（6指標）

入力人物のID(個人性)に対する忠実度

プロンプト(テキスト)と出力画像の一致度

IdentityとTextのバランス評価

Method	Identity-Fidelity (↑)			Text-Fidelity (↑)			Identity×Text (↑)	
	AdaFace	SphereFace	FaceNet	CLIP	dCLIP	SigLIP	hMean	gMean
Subject-driven								
TextualInversion [17]	0.1654	0.2518	0.3197	0.1877	0.1075	0.1283	0.0320	0.0575
DreamBooth [46]	0.3482(2)	0.4084	0.4774	0.2351	0.1480	0.3452(3)	0.0689	0.1116
CustomDiffusion [30]	0.4537(1)	0.5624(1)	0.6458(1)	0.2023	0.1490	0.2182	0.1078	0.1700(3)
Perfusion [54]	0.0925	0.1478	0.1887	0.2490	0.1686	0.3433	0.0342	0.0575
E4T [18]	0.0433	0.1190	0.1725	0.2510(2)	0.1784	0.2671	0.0370	0.0589
CelebBasis [63]	0.1601	0.2724	0.3762	0.2563(1)	0.1847(3)	0.3624(2)	0.1138(3)	0.1542
Subject-agnostic								
FastComposer [60]	0.1736	0.3271	0.4725	0.2495(3)	0.2149(1)	0.3168	0.1655(2)	0.2120(2)
ELITE [59]	0.0924	0.1925	0.3232	0.1755	0.1250	0.0671	0.0308	0.0624
DreamIdentity* [11]	0.2847	0.4252(2)	0.5523(2)	0.1924	0.1463	0.1539	0.0849	0.1326
Face2Diffusion (Ours)	0.3143(3)	0.4215(3)	0.5313(3)	0.2486	0.2020(2)	0.3856(1)	0.1749(1)	0.2252(1)

AdaFace	頑健な顔認識モデルによる類似度	Arc Face + adaptive margin	難例に強い	CLIP	元テキストと画像のCLIP類似度	最も基本的な整合性評価	hMean	調和平均：$\frac{2ab}{a+b}$
SphereFace	顔分類器ベースの特徴比較	球面損失に基づく顔認識	Angularな特徴に強い	dCLIP	変化ベクトルの方向が一致しているか	Text inversionの評価	gMean	相乗平均：$\sqrt{ab}$
FaceNet	三重項損失に基づくベクトル距離	Googleの有名モデル	汎用性が高いが古い	SigLIP	GoogleのSigmoid-based CLIP類似モデルでの整合性	よりリッチな視覚概念対応		

比較結果

結果の比較


Input S*
CustomDiffusion
CelebBasis
FastComposer
DreamIdentity
Face2Diffusion
"A photo of S* as a DJ"
"A photo of S* giving a thumbs-up"


Input S*
TextualInversion
DreamBooth
CustomDiffusion
Perfusion
E4T
CelebBasis
FastComposer
ELITE
DreamIdentity
Face2Diffusion
"A photo of S* as a DJ"
"A photo of S* walking in a city under an umbrella"
"A photo of S* trying on hats in a vintage boutique"
"A photo of S* sipping coffee at a café terrace"
"A photo of S* on a gondola in Venice"

比較結果

FID(画像の全体的品質と多様性を測る指標) + 学習パラメータ数 + エンコード速度

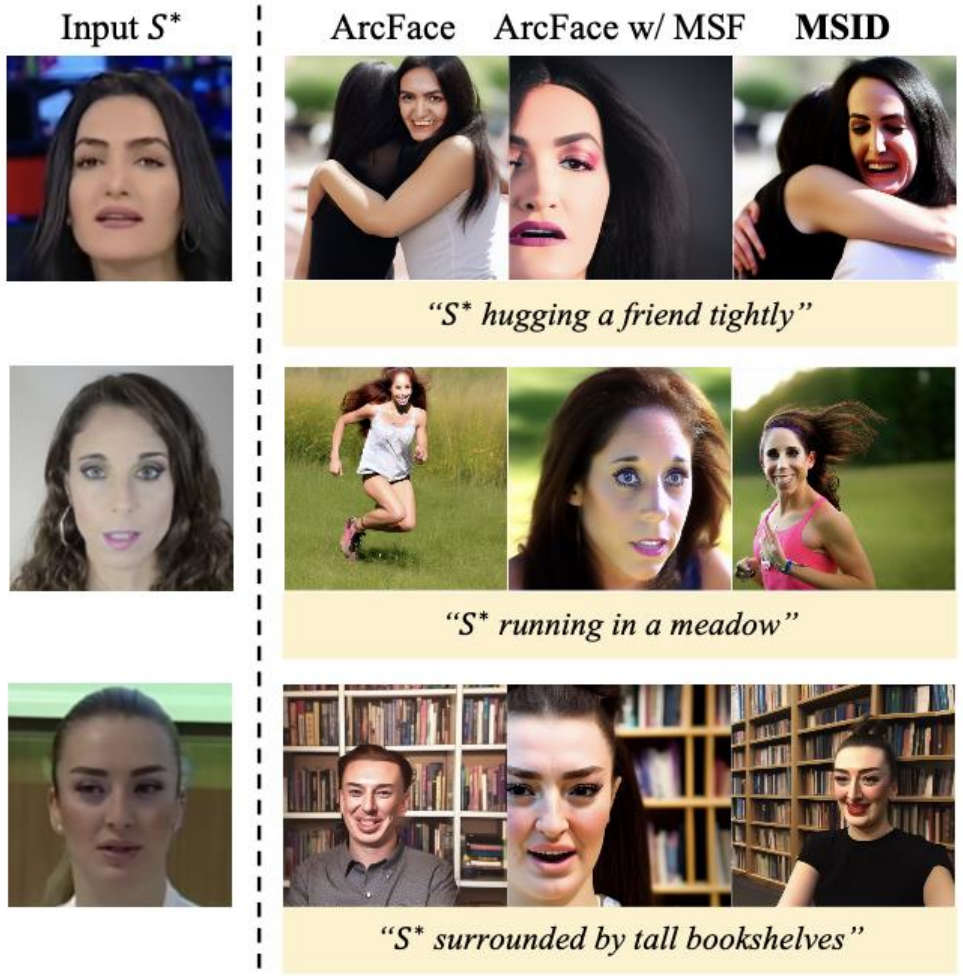
	CustomDiffusion	CelebBasis	FastComposer	Ours
FID (↓)	86.18	<u>69.87</u>	77.62	**69.33**
#Params (↓)	5.71×10^7	**1024**	8.88×10^8	<u>1.20×10^7</u>
Time (↓)	140 sec	220 sec	<u>0.026 sec</u>	**0.006 sec**

Table 2. **Comparison on FID and the computation costs.**

MSID Encoder の効果

Encoder	AdaFace	CLIP	hMean
ArcFace [12]	0.2421	**0.2758**	<u>0.2135</u>
ArcFace w/ Multi-Scale Feat. [11]	**0.3264**	0.2069	0.1549
MSID Encoder (Ours)	<u>0.3143</u>	<u>0.2486</u>	**0.2252**

ArcFace	Identityの類似度が低い	テキスト整合性が高い
ArcFace + Multi-Scale	Identityの類似度が高い	テキスト整合性が低い
MSID	どちらも真ん中	調和平均が最も高い



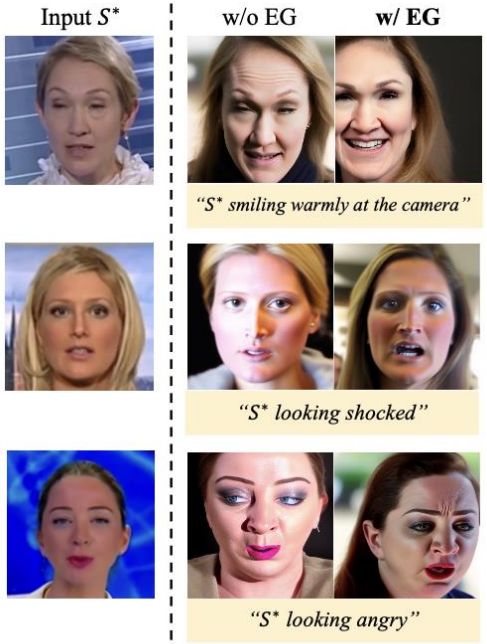
(a) Effect of MSID encoder.

比較結果

Expression Guidance の効果

Method	AdaFace	CLIP	hMean
w/o Expression Guidance	**0.3338**	0.2315	0.2032
w/ Expression Guidance (Ours)	0.3143	**0.2486**	**0.2252**

Table 4. **Effect of expression guidance.**



(b) Effect of expression guidance.

CGDR(Class-Guided Denoising Regularization)

Method	AdaFace	CLIP	hMean
Reconstruction	0.3133	0.2053	0.1411
Masked Reconstruction	0.3012	0.1990	0.1282
Reconstruction w/ DSC [60]	0.1651	**0.2851**	0.1596
CGDR (Ours)	**0.3143**	0.2486	**0.2252**

Reconstruction Identity 類似度が一定高い

Masked Reconstruction 特にいいとこなし

Reconstruction DSC text整合度は高いがidentity類似度が致命的に落ちる

CGDR Identity 類似度とバランス

w/o Expression Guidance Identityの類似度が高い

w/ Expression Guidance Identityの類似度が比率分減少

(c) Effect of CGDR.

比較結果

Upper-bound Analysis

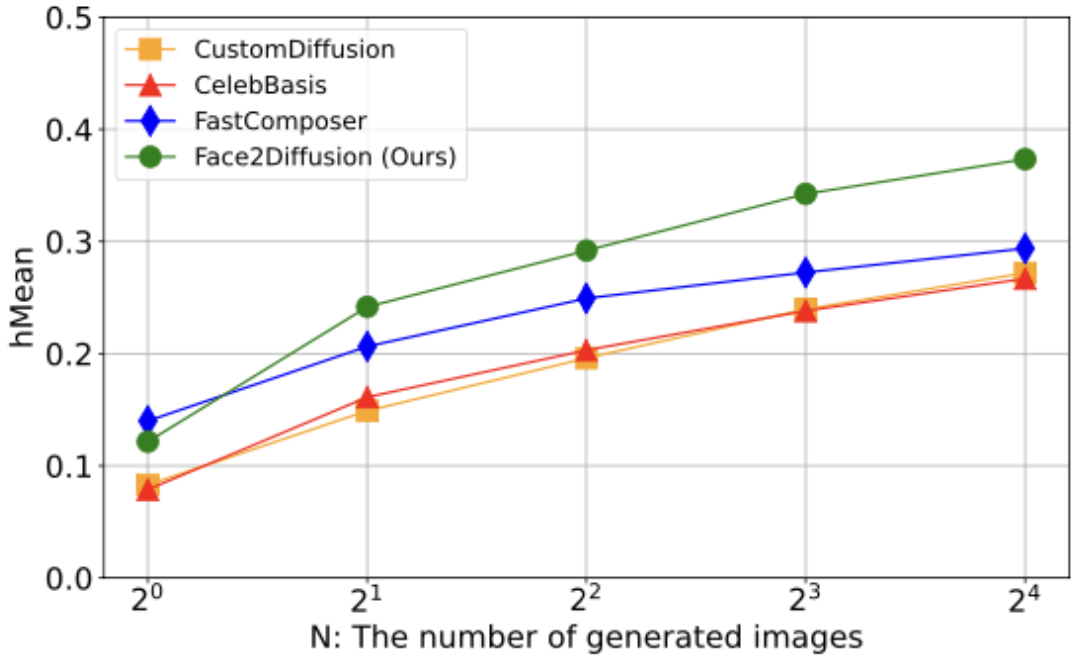


Figure 7. **Upper-bound analysis.**

Face 2 Diffusion：
ガウスノイズを付与する回数が多くするにつれて、
他のモデルと比較して**identity と text 調和性**の**調和平均が上昇**することがわかる

6 : Conclusion

結論

【有用性】

既存の拡散モデルに簡単に統合可能な顔ID保持モジュールの開発

1. **Multi – Scale identity Encoder (MISD)**
   - **角度の内積計算**によって類似度を算出し、浅層での識別を強化
2. **Expression Guidance**
   - **3D表情特徴を導入**し、identity を損なうことなく表情の制御性を実現
3. **Class-Guided Denoising Regularization（CGDR）**
   - **背景と顔を空間的に分離**して制御。背景の過学習を防ぎつつ、多様な生成を可能に

【Face 2 Diffusion の限界】

1. **オープンワールドへの汎化に限界**
   - モデルは主に**FFHQやセレブ画像に基づいて**訓練されており、極端な顔の特徴（例：アニメ、特殊メイク、非人間顔）には弱い
2. **対象人物の identity embedding に依存**
   - 提案手法では、identity情報が正しく埋め込まれている前提があるため、**元画像が不鮮明／低品質**であると性能が落ちる
3. **背景・表情の精度はプロンプトとマスクに依存**
   - 背景生成や表情制御は、**マスク(顔領域)やCLIPプロンプト**の**表現力**に影響を受けやすい

今後の課題・所感

【Face 2 Diffusion の限界】

オープンワールドへの汎化に限界

モデルは主に**FFHQやセレブ画像に基づいて**訓練されており、極端な
顔の特徴（例：アニメ、特殊メイク、非人間顔）には弱い

対象人物の identity embedding に依存

提案手法では、identity情報が正しく埋め込まれている前提があるため、
元画像が不鮮明／低品質であると性能が落ちる

背景・表情の精度はプロンプトとマスクに依存

背景生成や表情制御は、**マスク（顔領域）やCLIPプロンプト**の**表現力**に影響
を受けやすい

【所感】

- **2020～2025年で凄まじい速度で研究されていて驚愕した（git-hubをみんな公開しているからかも）**
- **そもそもの拡散モデルの概念は理解していたつもりだったが、ここまで体系化して理解できてよかった**
- **自分の研究にも活かしたいが、Face2Diffusionの枠組みをそのまま使用するのは難しそうだけど、拡散モデルは活かせそう**
  /拡散モデルの概念を抽象化すると、高次元データを低次元データに落としながら学習して、再び近しい高次元データに戻す作業
  /顔に関しては人間の脳による補完能力がかなり寄与している気がする/ミクロで見ると正確ではないが、マクロではかなり似てる　そんな感じ
- **都市計画の文脈での研究**
  - **GPSの軌跡データのプライバシー保護と、高品質な再生性　　etc...　次のページ**

拡散モデルを用いた都市計画領域での研究

論文名	著者名	発行年	概要
DiffTraj: Generating GPS Trajectory with Diffusion Probabilistic Model	Ziyi Liu et al.	2023	拡散モデルを用いてプライバシー保護されたGPS軌跡データを生成。
CoDiffMob: Diffusion Model-based Urban Mobility Generation with Collaborative Noise Priors	Jingtao Ding et al.	2024	集団的・個人的特性を反映した都市移動データ生成の手法を提案。
TrajGDM: Simulating Human Mobility with a Trajectory Generation Framework Based on Diffusion Model	Sheng Wang et al.	2024	拡散モデルを使い、人の移動軌跡をシミュレーション・予測する手法を提示。
UP-Diff: Urban Planning Prediction with Remote Sensing via Latent Diffusion Models	Siyu Zhai et al.	2024	潜在拡散モデルを用いて都市の将来の土地利用予測を行う。
LayoutDM: Discrete Diffusion Model for Controllable Layout Generation	Xingran Chen et al.	2023	制約条件下でレイアウトを生成する離散拡散モデルの提案。

拡散モデルを用いたGPSデータ関係の研究

論文名	著者名	発表年	概要
Diffusion Models for Privacy-Preserving Synthetic Mobility Data Generation	Song et al.	2023	モビリティデータのプライバシー保護と再生成を拡散モデルで実現する手法を提案。
Differentially Private Trajectory Data Publishing Using Generative Models	Zhang et al.	2022	差分プライバシーと生成モデルを組み合わせ、GPSデータの匿名化を実現。
Diffusion Models for Human Mobility Patterns Reconstruction	Wang et al.	2023	人間の移動パターンを拡散モデルで再構成し、リアルな軌跡を生成する研究。

拡散モデルを用いたGPSデータ関係の研究

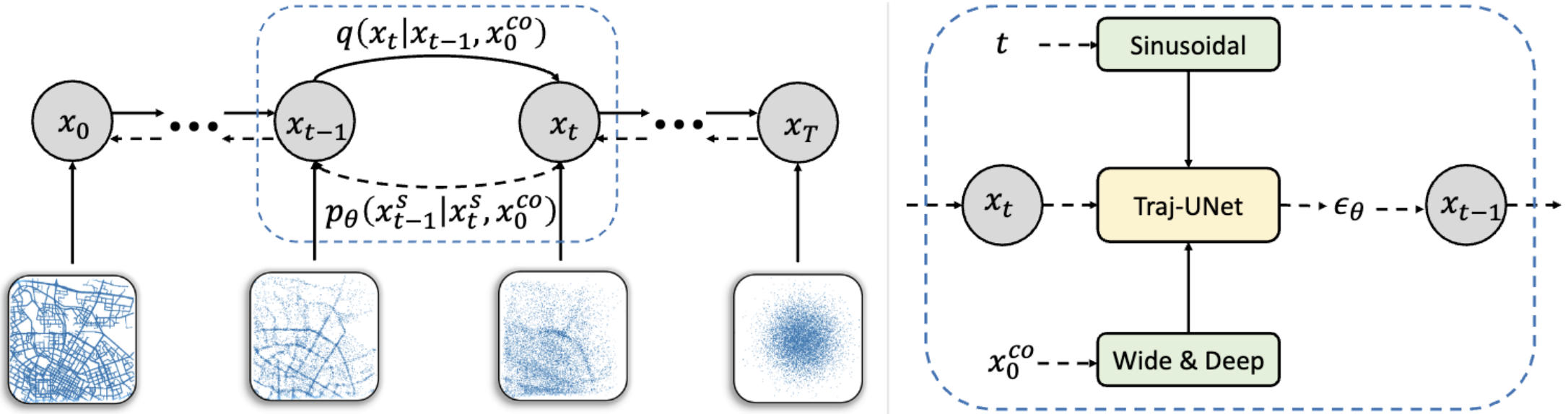


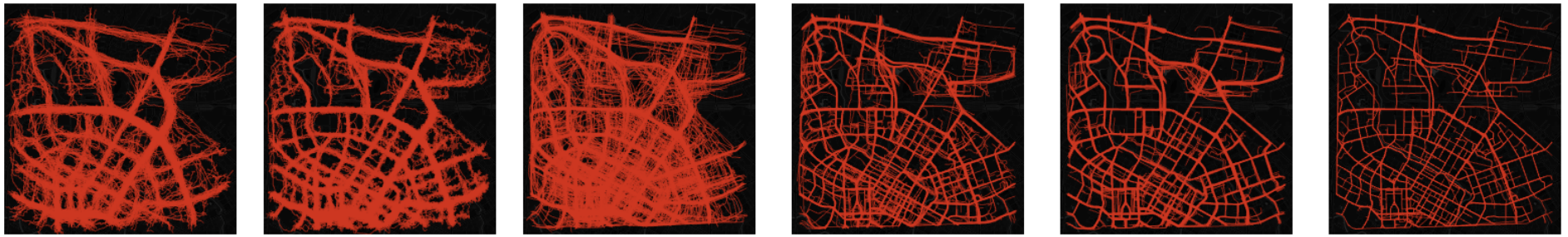
Figure 1: An illustration for trajectory generation with diffusion model. (Left) Forward and the reverse process (multiple GPS trajectories presented). (Right) Coupled the neural network model structure for reverse denoising.

拡散モデルを用いたGPSデータ関係の研究

Table 1: Performance comparison of different generative approaches.

Methods	Chengdu				Xi'an			
	Density (↓)	Trip (↓)	Length (↓)	Pattern (↑)	Density (↓)	Trip (↓)	Length (↓)	Pattern (↑)
RP	0.0698	0.0835	0.2337	0.493	0.0543	0.0744	0.2067	0.381
GP	0.1365	0.1590	0.1423	0.233	0.0928	0.1013	0.2164	0.233
VAE	0.0148	0.0452	0.0383	0.356	0.0237	0.0608	0.0497	0.531
TrajGAN	0.0125	0.0497	0.0388	0.502	0.0220	0.0512	0.0386	0.565
DP-TrajGAN	0.0117	0.0443	0.0221	0.706	0.0207	0.0498	0.0436	0.664
Diffwave	0.0145	0.0253	0.0315	0.741	0.0213	0.0343	0.0321	0.574
Diff-scatter	0.0209	0.0685	–	–	0.0693	0.0762	–	–
Diff-wo/UNet	0.0356	0.0868	0.0378	0.422	0.0364	0.0832	0.0396	0.367
DiffTraj-wo/Con	0.0072	0.0239	0.0376	0.643	0.0138	0.0209	0.0357	0.692
Diff-LSTM	0.0068	0.0199	0.0217	0.737	0.0142	0.0195	0.0259	0.706
DiffTraj	**0.0055**	**0.0154**	**0.0169**	**0.823**	**0.0126**	**0.0165**	**0.0203**	**0.764**

Bold indicates the statistically best performance (i.e., two-sided t-test with $p < 0.05$) over the best baseline.



(a) VAE (b) TrajGAN (c) Diffwave (d) Diff-LSTM (e) DiffTraj (f) Real

拡散モデルを用いたGPSデータ関係の研究

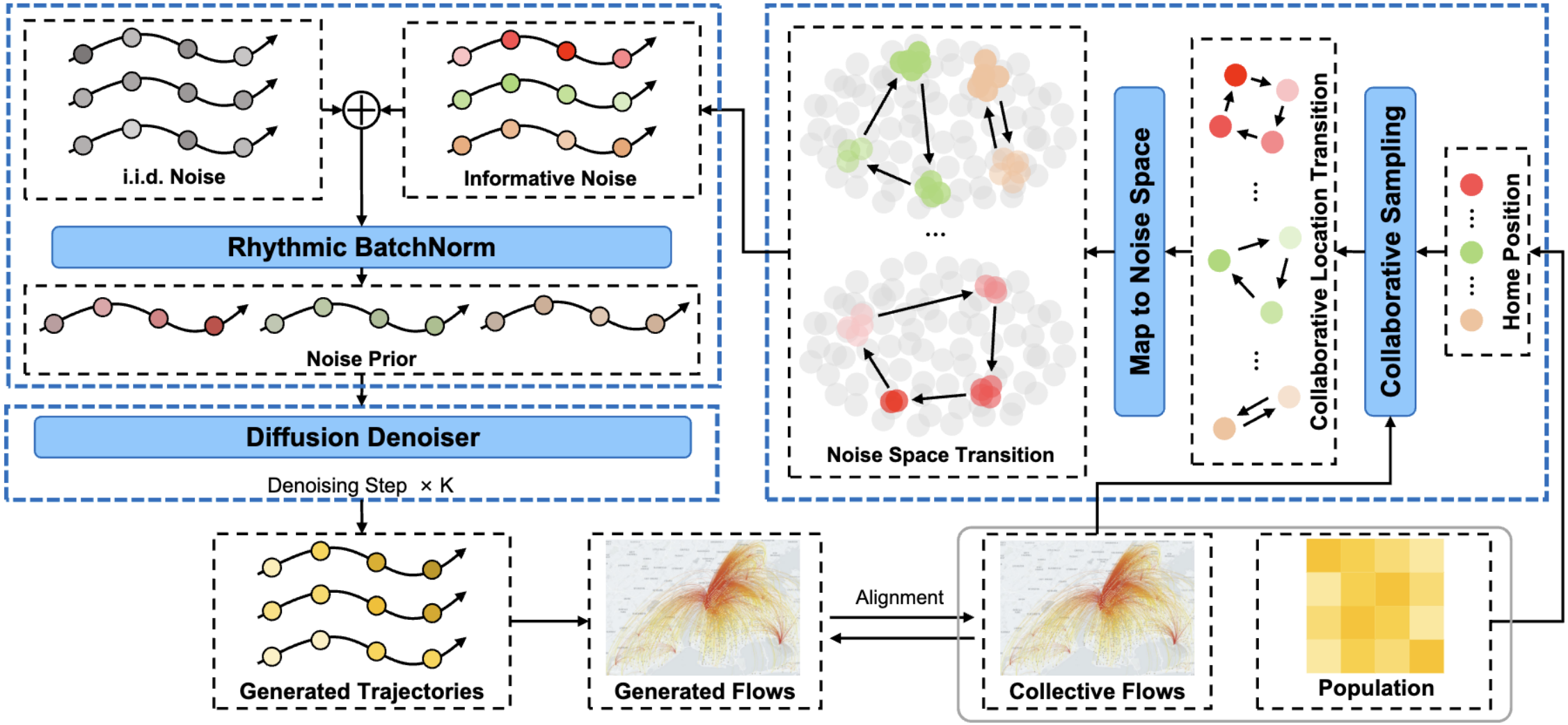


Figure 2: The overall workflow of mobility generation with collaborative noise priors.

拡散モデルを用いたGPSデータ関係の研究

Dataset	ISP							MME						
Metrics	Trajectory				Flow		Diversity	Trajectory				Flow		Diversity
	Radius	Distance	Duration	DailyLoc	CPC	MAPE		Radius	Distance	Duration	DailyLoc	CPC	MAPE	
TimeGEO	0.3171	0.2984	0.0526	0.2877	0.2133	0.8637	0.1498	0.3678	0.4807	0.0891	0.2512	0.3248	0.8274	0.1682
PateGail	0.1979	0.1826	0.1867	0.2933	0.0714	0.9827	0.0984	0.1824	0.1673	0.1376	0.2883	0.1548	0.9136	0.1136
MoveSim	0.2313	0.2623	0.2861	0.4289	0.0932	0.9983	0.0492	0.1936	0.2219	0.2084	0.3746	0.1739	0.8873	0.0492
VOLUNTEER	0.4435	0.4716	0.4245	0.5155	0.0044	1.0088	0.0839	0.3826	0.3911	0.3512	0.4558	0.0927	0.9512	0.0928
TrajGDM	0.2977	0.2393	0.2443	0.4121	0.0070	1.0022	0.1271	0.3123	0.2577	0.2875	0.3961	0.0526	0.9813	0.1474
DiffTraj	0.0519	0.1662	0.0895	0.1617	0.3992	0.8331	**0.0028**	0.1028	0.1242	0.0949	0.2657	0.4198	0.7727	**0.0053**
CoDiffMob	**0.0183**	**0.1203**	**0.0245**	**0.1558**	**0.6842**	**0.6361**	0.0046	**0.0519**	**0.0970**	**0.0796**	**0.1729**	**0.6067**	**0.6731**	0.0089

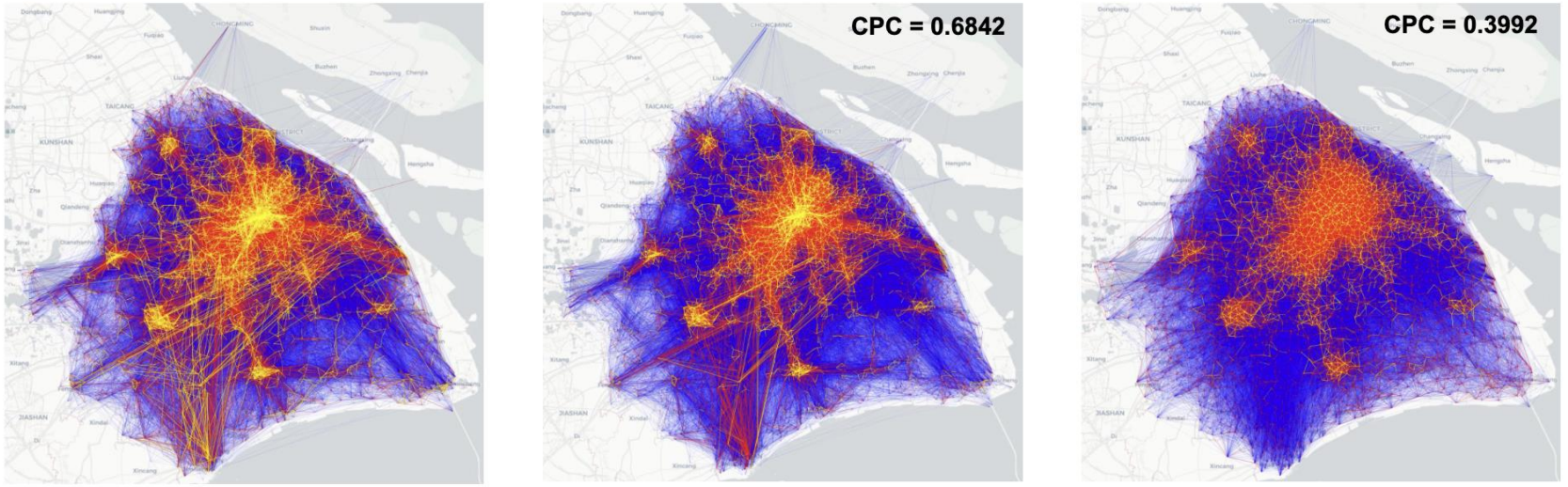


Figure 3: Visualization of the observed flows and generated flows on ISP dataset.

7：オリジナル　ケーススタディ

Original Case Study

Input S*

 # 入力プロンプト

conda run -n f2d python3 inference_f2d.py /

-p “ 【f l】=S* --------- ---- ”/ **←入力プロンプト**

-i input/Hato_ver2_rgb.png / **←画像名**

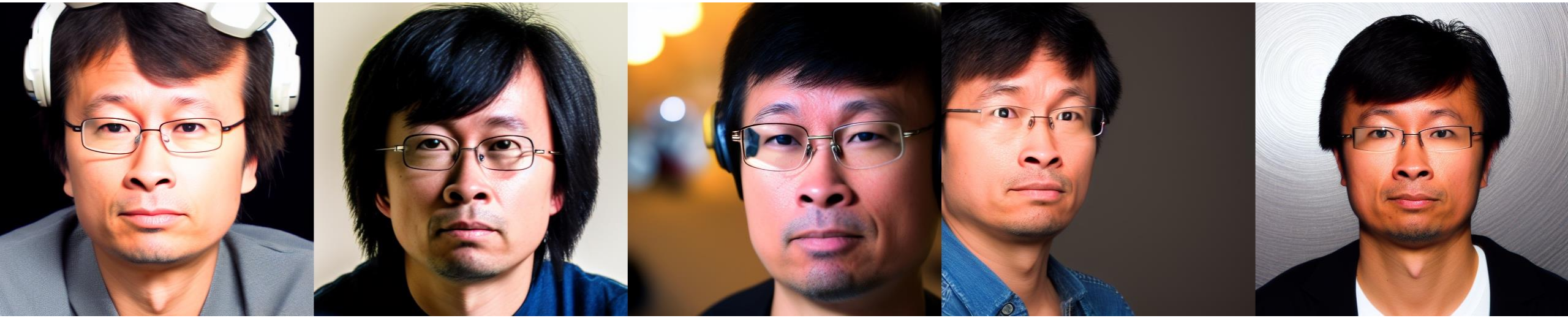
--w_map checkpoints/mapping.pt / **← Mapping Network用のモデル**

--w_msid checkpoints/msid.pt / **← Multi-Scale Identity Encoder 用のモデル**

-o output_professor_ver2.png /

-n 10 **← 生成枚数**

Original Case Study



"f l as a DJ "

Original Case Study

"f l is working as a DJ at the pool side"

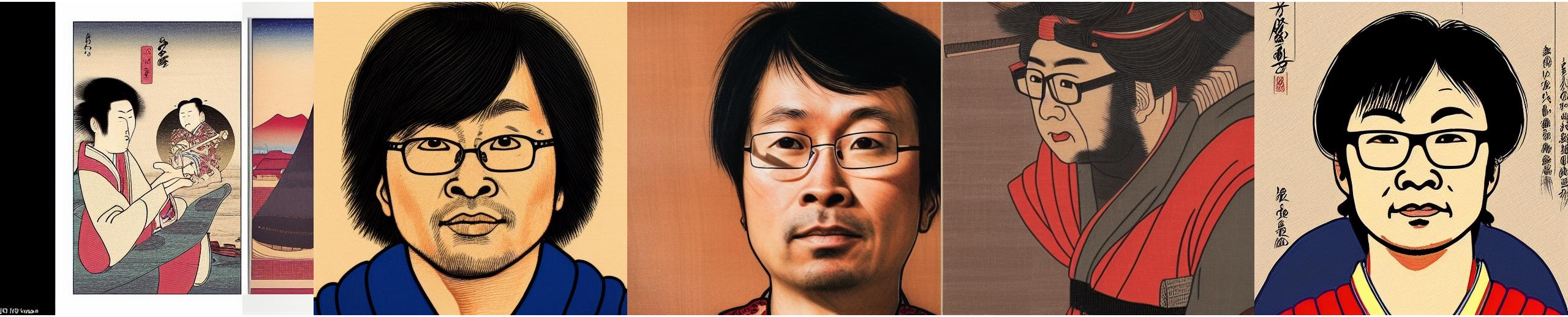
Original Case Study

"f l is working as a DJ at the bar"

Original Case Study

"f I looking happy, he is working as a DJ at the bar"

Original Case Study



f l is a man,
in the style of Japanese Ukiyo-e, bold woodblock lines, flat vivid colors, traditional Edo-period scenery

Original Case Study

"f l is portrayed like a historical genius such as Beethoven,
with a serious expression, painted in the style of a classical oil portrait, dark background,
inside a golden ornate frame, realistic lighting, 19th century style"

7 : Appendix

研究の発展

拡散モデルの発展

拡散モデルのこれまで

年代	著者／団体	タイトル	アプローチ	備考
2023	Rombach et al.	Stable Diffusion 2.0	潜在拡散モデルの改良版VAEの再設計、新しいテキストエンコーダ、改善されたノイズスケジュールの導入	オープンソースで公開
2023	Stability AI	Stable Diffusion XL（SDXL）	“XL”アーキテクチャによる大規模潜在拡散＆より深いU-Net構造 + CLIP-Fine-Tuning手法を採用し、テキスト／画像品質を同時に強化	768×768 → 1024×1024 超えの高解像度生成が可能　ノイズ除去能力と細部表現が大幅改善
2023	OpenAI	DALL·E 3	GPT-4系列言語理解を拡散モデルに組み込み、GPTで適切なプロンプト補完→改良型Diffusionステップを経て画像を出力	ChatGPT連携でプロンプト作成・修正を対話的にサポート　DALL·E 2よりテキスト整合性が大幅向上
2023	Saharia et al. (Google Research)	Imagen 2	大規模言語モデル（LaMDA系列）強化 + マルチスケール注意機構を組み合わせた拡散ネットワークで、文脈理解力と出力画像品質を両立	複数モダリティ統合可能で先行のImagenより精緻な画像生成
2023	Midjourney Team	Midjourney v5	独自潜在拡散ベース + 拡張カスタム・パラメータ空間で多彩なスタイルやシーンを高忠実度生成、階層型CLIPガイダンスで細部を制御	論文未公開だがコミュニティで高評価 複雑なプロンプト耐性・絵画的表現の豊かさが特徴
2023	Ho et al. (DeepMind)	DiT (Diffusion Transformer)	**Transformerアーキテクチャをノイズ予測に直接適用し、ピクセル空間拡散ステップと自己注意を組み合わせたDiffusion + Transformerハイブリッド**	ImageNet高解像度生成でSOTA更新 従来U-Net系より計算リソース高いが、一貫性とシャープさに優れる
2024	Rombach et al.	Stable Diffusion 2.1	Stable Diffusion 2系列のマイナーアップデート。新テキスト条件付けモジュール（改良版CLIP + 可逆正規化）で物体の細部・文字能力を強化	SD2.0の文字歪み問題を多く解消 実運用環境での安定性向上
2024	Google Research	Imagen 3	マルチモーダル事前学習を拡張し、テキストだけでなく画像中オブジェクト・構造知識を反映する拡散ネットワークを提案。高速・高解像度生成を実現	Imagen系列を超えるスケーラビリティと複雑シーン生成能力獲得 映画品質フレーム生成など応用想定
2024	Runway ML	Runway Gen-2	画像→動画拡張版拡散モデル 静止画を拡散で生成/編集 → 時系列情報考慮の2段階目拡散で自然な連続フレームを作成	動画クリップ・アニメーション生成に適用可能 従来画像拡散モデルよりTemporal Consistency重視

研究の発展 テキスト→画像生成 のこれまで

GLIDE(OpenAI)
テキスト→画像拡散モデル
ノイズから画像を徐々に生成すると同時に
生成過程にテキスト条件を組み込む
「分類器フリーガイダンス」を活用

Imagen(Google)
テキスト→画像拡散モデル
大規模言語モデル ＆ 拡散モデル
リアルな描写と深い言語理解を両立

Textual Inversion
事前学習済みの潜在拡散モデルの「単語」を学習
個別の概念(個人の小物やスタイルなど)を
テキスト埋め込みとして表現
テキストプロンプトを自然に組み込めるように
するパーソナライゼーション手法

DALL·E 2 (OpenAI)
第2世代のテキスト→画像モデル
テキストからオリジナルかつリアルな画像やアートを生成
複数の概念や属性,スタイルを組み合わせた合成が可能
インペインティングやアウトペインティング機能も備える

Stable Diffusion(Stability AI)
オープンソースの拡散モデルファミリー
最大1メガピクセル解像度での高品質生成が可能
セルフホストやクラウド統合など幅広い運用形態

DreamBooth
少数のリファレンス画像のみを使用
テキスト→画像拡散モデルをファインチューニング
特定の被写体（人や物）の識別子と結びつけて、多様な
シーンや視点で被写体を高忠実度に再現

研究の発展 テキスト→画像生成 のこれまで

CLIP(Open AI)
コントラスト学習を用いたマルチモーダルモデル、
大規模(画像, テキスト)ペアを用いて自然言語と
画像の表現空間を同時に学習し、ゼロショットで
の概念認識や指示に従った処理

ELITE
大規模事前学習モデルをチューニング不要で活用し、
Open Images データセット上でマッピングネット
ワークとクロスアテンション層を二段階学習する戦略
テキスト→画像パーソナライゼーションを実現

Celeb Basis
有名人名の基底から顔埋め込みを構築
個別の顔表現を生成 T2I モデルに人物を組み込む
基底ベクトルを組み合わせて多様な顔表現を可能
にする

Fast Composer
ローカライズされたクロスアテンション機構を用い、
複数人物が混在するシーンでも各人物を正確に配
置・生成できるようにしたテキスト→画像生成手法

式の補足説明

（Jensen の不等式）
ある確率変数 X と、φ が**凹（concave）関数**であれば、

$$\varphi\big(\mathbb{E}[X]\big) \;\geq\; \mathbb{E}\big[\varphi(X)\big].$$

「φ が凸（convex）関数」であれば不等号は逆向きになります：

$$\varphi\big(\mathbb{E}[X]\big) \;\leq\; \mathbb{E}\big[\varphi(X)\big].$$

ここで、定番としてよく使うのが「$\varphi(t) = \log t$」の場合です。
$\log(\cdot)$ は凹関数なので、

$$\log\big(\mathbb{E}[Y]\big) \;\geq\; \mathbb{E}\big[\log Y\big] \quad (\text{ただし } Y > 0)$$