Multimediaゼミ 2024年9月24日

- 渡部
  - SwiftBrush：One-Step Text-to-Image Diffusion Model with Variational Score Distillation (https://arxiv.org/abs/2312.05239)
  - 従来の拡散モデルの蒸留方法は、膨大な画像データを学習中に必要としていた。そこで画像データを必要としない蒸留方法SwiftBrushを提案。訓練用の画像データを使用せずに、Stable Diffusionと同等の品質の画像を生成することができるワンステップのtext-imageモデルを作成することができた
  - 関連手法
    - Score Distillation Sampling
      - この手法では、パラメータ θ で表される単一の3D NeRFを、与えられたテキストプロンプトに一致するよう最適化する。カメラ視点 c が与えられた場合、微分可能なレンダリング関数 g(·, c) を使用して、3D NeRF からカメラ視点 c における画像をレンダリングする。ここで、レンダリングされた画像 g(θ, c) は、勾配が式 4 で近似できる損失関数を通じて重み θ を最適化するために利用される
    - Variational Score Distillation
      - この手法では、カメラ位置 c における3D NeRFからレンダリングされた画像に特化した追加のスコア関数を導入することで、SDSと差別化している。このスコアは、式6の損失を最小化することによって拡散モデルをファインチューニングさせることで実現している
      - ϵ_ϕはLow-Rank Adaption（LoRA）によってパラメータ化され、カメラ視点 c を条件付けするための追加のレイヤーを持つ事前学習済みの拡散モデルϵ_ψ から初期化される
  - 提案手法
    - VSDにおいてNeRFをtext-imageモデルに置き換える

        - 2種類の教師モデルを使用 - 学習済みのtext-imageモデルとLoRAモデル
            - LoRAについてはカメラ視点cは取り除かれている
        - 生徒モデルはガウシアンノイズとテキストプロンプトが入力
        - LoRAと生徒モデルは学習済みの教師モデルのパラメータによって初期化されている
        - LoRAと生徒モデルを式5,6によって両方学習させる、その際には教師モデルのtext-imageモデルのパラメータはフリーズさせておく
            - LoRAの学習と生徒モデルの学習は交互に行う
        - 学習済みの教師モデル(Stable Diffusion)を使用するにあたって注意が必要になる。それは、Stable Diffusionは加えられたノイズを学習するのに対して、生徒モデルの目的はノイズから綺麗な画像を生成するのが目的であるため
            - そこでre-parameterizationによって、生徒モデルのアウトプットがノイズになるように調整
    - 結果

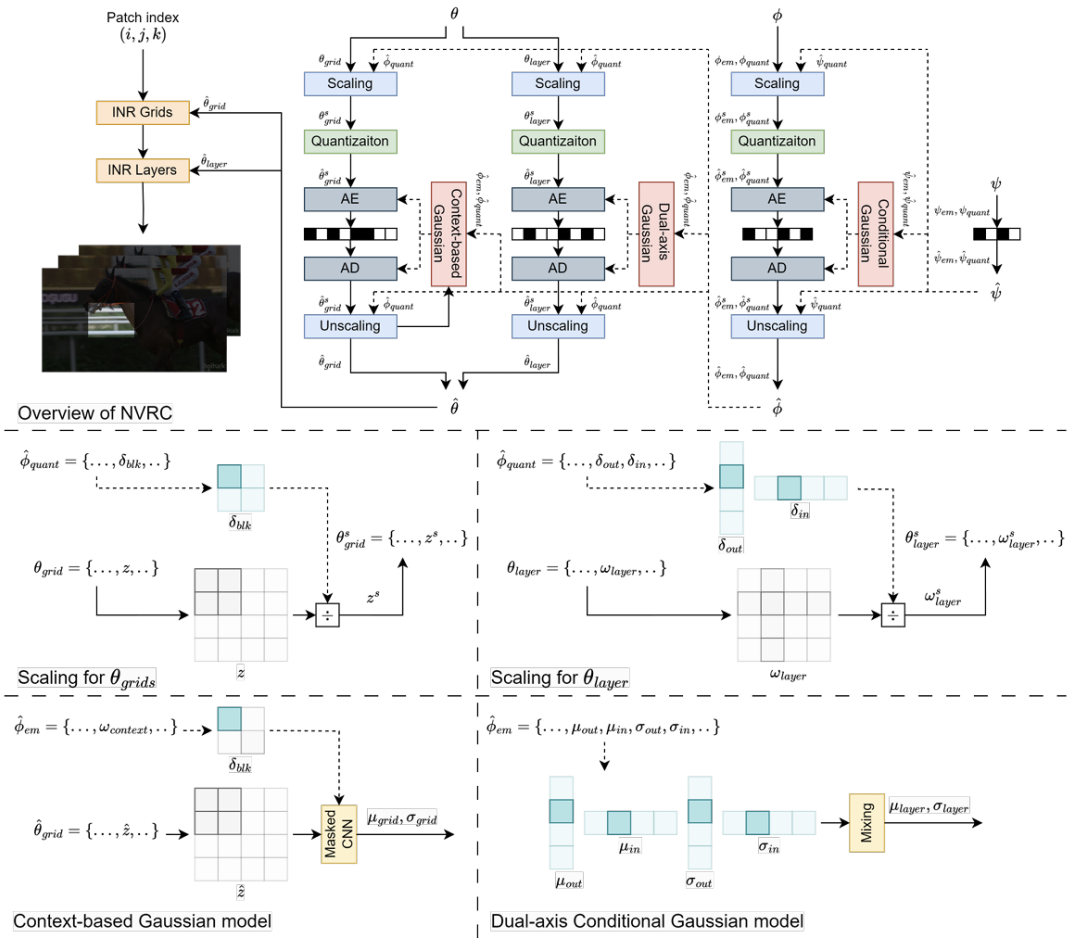
Method	Steps	FID-30K ↓	CLIP-30K ↑
Guided Distillation†	1	37.3	0.27
LCM†	1	35.56	0.24
Instaflow	1	**13.10**†	0.28§
BOOT‡	1	48.20	0.26
Ours	1	16.67	**0.29**
SD 2.1*	25	13.45	0.30
SD 2.1*	1	202.14	0.08

Table 1. Comparison of our method against other works based on FID metric and CLIP score on the COCO 2014 dataset. † means that we obtain the numbers from the corresponding papers. § means that we obtain the numbers using the provided pretrained models of the corresponding papers. ‡ means that we re-implement the work and report the numbers. * means that we report the number using the pretrained models. We use DPM-Solver [21] with guidance scale 7.5 to sample for SD. **Bold** and underlined numbers are best and second best for one-step models.

Models	Human Preference Score v2 ↑			
	Anime	Photo	Concept Art	Paintings
LCM†	22.61	22.71	22.74	22.91
InstaFlow†	25.98	26.32	25.79	25.93
BOOT‡	25.29	25.16	24.40	24.61
Ours	**26.91**	**27.21**	**26.32**	**26.37**
SD 2.1*	27.48	26.89	26.86	27.46

Table 2. Comparison of our method against other works based on HPSv2 score in 1-step regime. † means that we obtain the score using the provided pretrained models of the corresponding papers. ‡ means that we re-implement the work and report the score by ourselves. * means that we obtain the score from [47]. **Bold** and underlined numbers are best and second best, respectively.

- 速水
  - NVRC:Neural Video Representation Compression (https://arxiv.org/abs/2409.07414)
    - ネットワークの埋め込みに焦点を当てている
    - Video Overfit → Model Pruning → Model quantization → weight encoding
    - INIRを圧縮するための拡張フレームワークを提案



- INR Grids - 特徴量、INR Layers - ネットワーク
  - GridとLayerで圧縮方法を変える
    - Feature grid coding
      - ブロック分割により異なる量子化パラメータを適応
      - 自己回帰型のコンテキストモデルを使用して符号化
    - Network layer parameter coding
      - 同じ行・列のパラメータ間で量子化とエントロピーモデル
  - Rate-distortion optimization
    - 交互最小化
      - K+1ステップ毎に最初のKステップでDを最小化、K+1でRを最小化
    - 2段階のトレーニング

- C3で出てきたような方法

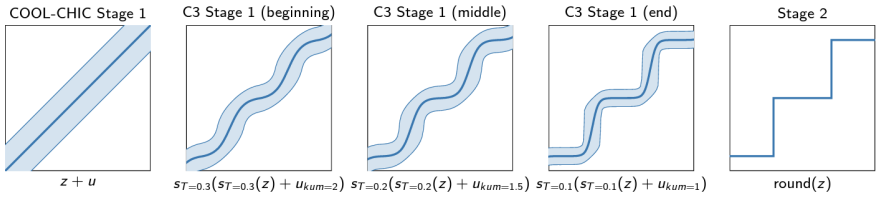


Figure 3. Approximating the $\mathrm{round}(\mathbf{z})$ function during Stage 1 of optimization. COOL-CHIC adds uniform noise $\mathbf{u}$, whereas C3 uses soft-rounding s_T with varying temperatures T and Kumaraswamy noise of different strengths, $\mathbf{u}_{\mathrm{kum}}$. We plot the mean and 95% interval.

- 結果

Table 1: BD-rate results on the UVG dataset.

Color Space	Metric	x265 (*veryslow*)	HM (*RA*)	VTM (*RA*)	DCVC-HEM	DCVC-DC	HiNeRV	C3	HNeRV-Boost
RGB 4:4:4	PSNR	-74.02%	-51.00%	-24.34%	-41.30%	-32.05%	-50.73%	-67.93%	-66.78%
	MS-SSIM	-80.79%	-67.61%	-50.08%	-7.91%	-12.58%	-44.69%	-	-78.21%
YUV 4:2:0	PSNR	-62.71%	-34.83%	-1.03%	-	-62.28%	-	-	-
	MS-SSIM	-59.49%	-38.45%	-15.38%	-	-70.23%	-	-	-

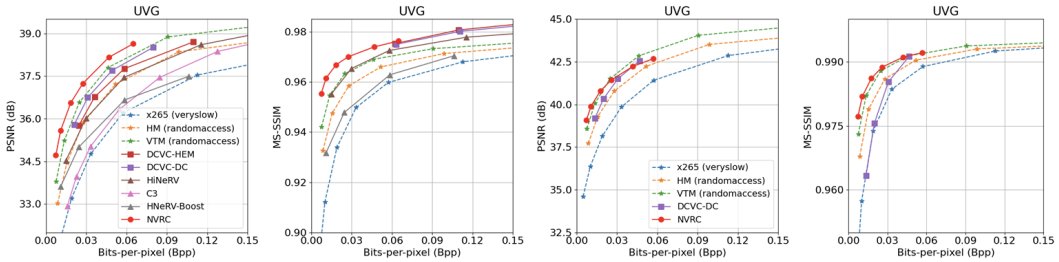


Figure 3: Average rate quality curves of various tested codecs on the UVG datasets.

- minghao
  - Speech Foundation Model Ensembles for the Controlled Singing Voice Deepfake Detection (CtrSVDD) Challenge 2024 (https://arxiv.org/abs/2409.02302)
    - AIによって生成された音声かどうかを判別するチャレンジ
    - 音声の基盤モデルを活用して、堅牢な歌声のアンチスプーフィングシステムを開発するためのアンサンブル手法を提案。また、音声の基盤モデルからの表現特徴を効率的かつ効果的に統合し、他の個々のシステムの性能を上回る新しい「Squeeze-and-Excitation Aggregation（SEA）」手法を提案

- 提案
    - データ拡張
        - RawBoost Augmentation
            - 線形と非線形の畳み込みのノイズ
            - ランダムな割合で信号にノイズを加える
            - 信号全体に均一に加えられる定常的な信号非依存ノイズ
    - 音声 → SSL Frontend → Layer Aggregation → Backend → Classifier → Predicted Score
        - Frontend
            - Raw waveform
                - フロントエンドとして70フィルターを持つRawNet2スタイルの学習可能なSincConv層を使用した。これらのSincConv層は、生の音声信号から重要な特徴を効果的に捉えるように設計されており、後続のタスクのために音声データを処理および分析するモデルの能力を向上させる
            - wav2vec2
                - 生の音声入力から直接幅広い音声特徴を効果的に捉える
            - WavLM
                - 話者の識別、パラ言語学、および発話内容を含む、音声信号の多面的な特性に対応するための大規模な事前学習済み音声基盤モデル
        - Layer Aggregation
            - Weighted Sum
                - 調整可能なパラメータを使用して、複数のニューラルネットワーク層からの出力を組み合わせる
            - Attentive Merging (AttM)
                - 時間次元にわたって埋め込みを平均化し、隠れ次元を圧縮するために全結合層を適用することで、アンチスプーフィ

ングにおいて最も関連性の高い特徴を強調する

- パラメータ数が多い

- Proposed SE Aggregation (SEA)
  - SEモジュールは、チャネル間の相互依存関係を明示的にモデル化することで、チャネルごとの関係をよくする。これにより、最も有益な特徴に焦点を当て、あまり有用でない特徴を抑制することで、ネットワークの表現能力が向上した

- Backend
  - The Graph Attention Layer (GAT)
    - ノード間のアテンションマップを計算し、アテンションメカニズムを用いてそれらを投影する
  - The Heterogeneous Graph Attention Layer (HtrgGAT)
    - スペクトル特徴ノードと時間的特徴ノードの両方を処理する
- Classifier
  - 分類器は、バックエンドモデルから抽出された特徴を利用し、その後の分類タスクを実行することで最終的な予測を出力する

- モデルアンサンブル
  - ここのモデルのアウトプットの平均を取る

- 結果

Index	Frontend	Layer Aggregation	Augmentation	EER of Datasets		EER of Different Attack Types						Pooled EER[8]	
				m4singer	kising	A09	A10	A11	A12	A13	A14	A09-A14	A09-A13
B01 [10]	LFCCs	-	-	-	-	-	-	-	-	-	-	-	11.37
B02 [10]	Raw waveform	-	-	-	-	-	-	-	-	-	-	-	10.39
B01 [8]	LFCCs	-	-	-	-	5.35	2.92	5.84	29.47	3.65	24.00	16.15	-
B02 [8]	Raw waveform	-	-	-	-	6.72	0.96	3.59	26.83	0.95	19.03	13.75	-
B02†	Raw waveform	-	-	10.77	10.73	6.14	1.01	3.76	24.43	1.18	18.55	12.75	9.45
M1	wav2vec2	-	-	5.55	13.97	2.21	1.84	5.02	9.11	2.62	19.07	9.87	4.80
M2	wav2vec2	-	series: (1)+(2)	6.83	9.71	2.16	2.03	8.71	6.95	2.34	13.57	7.94	5.99
M3	wav2vec2	-	parallel: (1)+(2)	3.94	10.00	1.59	1.17	3.19	7.37	1.81	13.70	6.88	3.55
M4	WavLM	Weighted Sum	series: (1)+(2)	4.68	8.81	2.21	1.46	5.62	5.77	1.66	12.98	6.66	4.10
M5	WavLM	Weighted Sum	parallel: (1)+(2)	3.40	8.85	1.35	0.98	3.70	5.78	1.07	12.52	5.91	3.16
M6	WavLM	AttM [24]	series: (1)+(2)	4.72	11.47	1.68	1.29	6.44	6.44	1.51	14.67	7.63	4.26
M7	WavLM	AttM [24]	parallel: (1)+(2)	3.48	10.73	**1.19**	**0.72**	3.81	6.02	**0.87**	13.70	6.51	3.22
M8	WavLM	Proposed SEA	series: (1)+(2)	3.81	8.53	1.32	0.93	3.72	5.95	1.15	12.83	6.16	3.32
M9	WavLM	Proposed SEA	parallel: (1)+(2)	**2.84**	8.36	1.62	1.23	**2.35**	5.24	1.32	12.46	**5.66**	**2.70**
M10	WavLM	Proposed SEA	parallel: (1)+(2)	3.26	9.54	1.52	1.06	2.66	5.98	1.16	12.91	5.94	3.02
M11	WavLM	Proposed SEA	series: (1)+(3)	6.57	5.03	2.47	1.79	9.53	**5.10**	1.97	**12.35**	7.36	5.77
M12	WavLM	Proposed SEA	series: (2)+(3)	7.24	**5.00**	2.71	2.26	8.70	6.66	2.46	13.56	7.76	6.08

Table 3. Performance in EER (%) on the evaluation set of CtrSVDD for ensemble systems.

Index	Ensembling Details	Ensemble Adjustments	EER of Datasets		EER of Different Attackers						Pooled EER[8]	
			m4singer	kising	A09	A10	A11	A12	A13	A14	A09-A14	A09-A13
E1	M5 + M7 + M8 + M9 + M10	-	2.71	8.40	1.03	0.74	2.56	4.77	0.88	12.33	5.39	2.50
E2	M3 + M5 + M7 + M8 + M9 + M10	+M3	2.41	7.19	0.82	0.56	2.17	4.24	0.69	12.00	5.01	2.21
E3	M3 + M5 + M7 + M9 + M10	-M8	2.30	7.21	0.79	0.55	2.00	4.17	0.70	11.94	4.96	2.13
E4	M2 + M3 + M5 + M7 + M9 + M10	+M2	2.09	6.47	0.68	0.48	1.96	3.83	0.63	**11.80**	4.78	1.95
E5	M2 + M3 + M7 + M9 + M10	-M5	**1.93**	**6.02**	**0.58**	**0.44**	**1.67**	**3.82**	**0.56**	11.84	**4.76**	**1.79**

- 髙部
  - GaussianShader: 3D Gaussian Splatting with Shading Functions for Reflective Surfaces (https://arxiv.org/pdf/2311.17977)
    - 全ての複雑な反射を残差色項に入れながら、拡散色と直接反射を考慮する
    - 関連研究
      - Ref-NeRF
        - Shading function
          - 直接反射のみを考慮
      - 3Dガウシアンについてはこのshading functionを計算するための法線の計算が難しい
    - 提案手法
      - 色の計算、投影点の計算、3D共分散計算のうち色の計算に注目
      - 色の計算
        - 各ガウシアンに対してShading Attributesが割り当てられて、球面の粗さや法線から反射する色を計算する
        - 粗さや法線は学習するパラメータ

        - ある3Dガウシアンがあった時に、ある視点を入れてどう色が移るかを考える
    - 結果

Table 1. The quantitative comparisons (PSNR / SSIM / LPIPS) on NeRF Synthetic dataset [32].

	NeRF Synthetic [32]								
	Chair	Drums	Lego	Mic	Materials	Ship	Hotdog	Ficus	Avg.
				PSNR↑					
NeRF [32]	33.00	25.01	32.54	32.91	29.62	28.65	36.18	30.13	31.01
VolSDF [52]	30.57	20.43	29.46	30.53	29.13	25.51	35.11	22.91	27.96
Ref-NeRF [45]	33.98	25.43	35.10	33.65	27.10	29.24	37.04	28.74	31.29
ENVIDR [27]	31.22	22.99	29.55	32.17	29.52	21.57	31.44	26.60	28.13
Gaussian Splatting [21]	35.82	26.17	35.69	35.34	30.00	30.87	37.67	34.83	33.30
Ours	35.83	26.36	35.87	35.23	30.07	30.82	37.85	34.97	33.38
				SSIM↑					
NeRF [32]	0.967	0.925	0.961	0.980	0.949	0.856	0.974	0.964	0.947
VolSDF [52]	0.949	0.893	0.951	0.969	0.954	0.842	0.972	0.929	0.932
Ref-NeRF [45]	0.974	0.929	0.975	0.983	0.921	0.864	0.979	0.954	0.947
ENVIDR [27]	0.976	0.930	0.961	0.984	0.968	0.855	0.963	0.987	0.956
Gaussian Splatting [21]	0.987	0.954	0.983	0.991	0.960	0.907	0.985	0.987	0.969
Ours	0.987	0.949	0.983	0.991	0.960	0.905	0.985	0.985	0.968
				LPIPS↓					
NeRF [32]	0.046	0.091	0.050	0.028	0.063	0.206	0.121	0.044	0.081
VolSDF [52]	0.056	0.119	0.054	0.191	0.048	0.191	0.043	0.068	0.096
Ref-NeRF [45]	0.029	0.073	0.025	0.018	0.078	0.158	0.028	0.056	0.058
ENVIDR [27]	0.031	0.080	0.054	0.021	0.045	0.228	0.072	0.010	0.067
Gaussian Splatting [21]	0.012	0.037	0.016	0.006	0.034	0.106	0.020	0.012	0.030
Ours	0.012	0.040	0.014	0.006	0.033	0.098	0.019	0.013	0.029

- 小泉
    - Anomaly Detection via Reverse Distillation from One-Class Embedding (https://arxiv.org/abs/2201.10703)
    - 教師なし異常検知
        - 再構成方法
        - Embeddingを用いた手法
            - その中の蒸留を用いた手法を紹介
        - 提案手法
            - 学習時に教師モデルと生徒モデルを学習させる

- その後、異常な画像を入れた際に教師モデルと生徒モデルの特徴量に差が生じた場合に異常とする
- 学習方法
  - imagenetで学習済みのTeacher Encoder + Student Decoder
  - 教師で出した特徴量と生徒で出した特徴量に対して、HとWできってベクトルにする
    - これらのなす角が小さくなるように学習する
    - そうすると推論時に異常なものが入力された場合にはなす角が大きくなる
  - Bottleneck
    - 教師の特徴量をそのまま生徒に渡すと、生徒がよくなりすぎて異常な入力でもなす角が小さくなってしまう
    - そこでBottleneckを加えることによって渡す特徴量を圧縮している

- 結果

Image Size		128		256								
Category/Method		MKD [33]	Ours	GT [10]	GN [2]	US [4]	PSVDD [43]	DAAD [16]	MF [40]	PaDiM [8]	CutPaste [23]	Ours
Textures	Carpet	79.3	99.2	43.7	69.9	91.6	92.9	86.6	94.0	**99.8**	93.9	**98.9**
	Grid	78.0	95.7	61.9	70.8	81.0	94.6	95.7	85.9	96.7	**100**	**100**
	Leather	95.1	100	84.1	84.2	88.2	90.9	86.2	**99.2**	**100**	**100**	**100**
	Tile	91.6	99.4	41.7	79.4	**99.1**	97.8	88.2	99.0	98.1	94.6	**99.3**
	Wood	94.3	98.8	61.1	83.4	97.7	96.5	98.2	**99.2**	**99.2**	**99.1**	**99.2**
	Average	*87.7*	*98.6*	*58.5*	*77.5*	*91.5*	*94.5*	*91.0*	*95.5*	*98.8*	*97.5*	***99.5***
Objects	Bottle	99.4	100	74.4	89.2	99.0	98.6	97.6	99.1	**99.9**	98.2	**100**
	Cable	89.2	97.1	78.3	75.7	86.2	90.3	84.4	**97.1**	92.7	81.2	**95.0**
	Capsule	80.5	89.5	67.0	73.2	86.1	76.7	76.7	87.5	91.3	**98.2**	**96.3**
	Hazelnut	98.4	99.8	35.9	78.5	93.1	92.0	92.1	**99.4**	92.0	98.3	**99.9**
	Metal Nut	73.6	99.2	81.3	70.0	82.0	94.0	75.8	96.2	98.7	**99.9**	**100**
	Pill	82.7	93.3	63.0	74.3	87.9	86.1	90.0	90.1	93.3	**94.9**	**96.6**
	Screw	83.3	91.1	50.0	74.6	54.9	81.3	**98.7**	**97.5**	85.8	88.7	97.0
	Toothbrush	92.2	90.3	97.2	65.3	95.3	**100**	99.2	**100**	96.1	99.4	**99.5**
	Transistor	85.6	99.5	86.9	79.2	81.8	91.5	87.6	94.4	**97.4**	96.1	**96.7**
	Zipper	93.2	94.3	82.0	74.5	91.9	97.9	85.9	**98.6**	90.3	**99.9**	98.5
	Average	*87.8*	*95.4*	*71.6*	*75.5*	*85.8*	*90.8*	*88.8*	*96.0*	*93.8*	*95.5*	***98.0***
Total Average		*87.8*	*96.5*	*67.2*	*76.2*	*87.7*	*92.1*	*89.5*	*95.8*	*95.5*	*96.1*	***98.5***