Supplamentary material: Rethinking cluster-conditioned diffusion models for label-free image synthesis

Nikolas Adaloglou
Heinrich Heine University of Dusseldorf
adaloglo@hhu.de

Tim Kaiser
Heinrich Heine University of Dusseldorf
tikai103@hhu.de

Felix Michels
Heinrich Heine University of Dusseldorf
felix.michels@hhu.de

Markus Kollmann
Heinrich Heine University of Dusseldorf
markus.kollmann@hhu.de

Code. Code is available at https://github.com/HHU-MMBS/cedm-official-wavc2025.

A. Visualizations

In Fig. 2, we map the TEMI clusters to classes using the Hungarian mapping [9] on CIFAR100. For the mapping to be one-to-one, we set $C = 100$. We then generate samples using the same initial noise with both C-EDM and GT-conditioned EDM and visualize the first 20 clusters on CIFAR100. Given the same initial noise and the cluster that is mapped to its respective GT class, we observe a lot of visual similarities in the images, even though the two models (C-EDM and EDM) have different weights and have been trained with different types of conditioning.

In Figs. 3 and 4, we visualize C-EDM samples generated from the same initial noise on FFHQ-64 for diffusion models trained with varying cluster granularities. Each noise gets a condition sampled from $p(c)$. Similar to our quantitative analysis, the generated images from small cluster sizes are closer to the unconditional prediction. Finally, in Fig. 1, we visualize cluster-conditioned and unconditional FFHQ samples at $M_{img} = 100M$.

In Fig. 5, we visualize real training FFHQ images that are grouped in the same TEMI cluster using the DINO features. We visually identify groups with shared characteristics such as sunglasses, hats, beanies, pictures of infants, and pictures of speakers talking to a microphone. Finally, in Fig. 6 we provide a more detailed visual comparison of low and high confidence samples using C-EDM on CIFAR100.

B. Additional discussion points

B.1. FID and FDD across training iterations

In Fig. 7, we report FID and FDD across training using C-EDM with TEMI clusters. We notice that FID tends to saturate faster than FDD and fluctuates more between checkpoints. FDD keeps decreasing monotonically, with minimal fluctuation and always prefers the samples at $M_{img} = 200$. Since both metrics compute the Frechet distance, these tendencies can only be attributed to the supervised InceptionV3 features. Even though the study of generative metrics is out of the scope of this work and a human evaluation is necessary as in [13], we hope that our findings w.r.t. cluster-conditioning can facilitate future works.

B.2. Image synthesis beyond ImageNet.

ImageNet is currently the largest labeled public dataset, and a single experiment using a recent state-of-the-art diffusion model on ImageNet requires up to 4MWh at 512^2 resolution [8]. Based on our experiments, clusters match or outperform the human-derived labels on image generation by estimating the visual groups. Using the introduced upper bound, the search space of the visual groups is significantly reduced with minimal computational overhead, while no further hyperparameter tuning is required. Therefore, it allows future works to incorporate unlabelled data and experiment at scales beyond ImageNet while being sample efficient. Additionally, the sample efficiency compared to noisy or non-mutually exclusive labels could be investigated in future works.

C. Deep image clustering with TEMI

C.1. Intuition for γ

In the TEMI loss function, there are two parts inside the log sum: the numerator $\left(q_s^i(c|x)q_t^i(c|x')\right)^{\gamma}$ aligns the cluster assignment of a positive pair and is maximal when each individual assignment is one-hot. On the other hand, the denominator $\tilde{q}_t^i(c)$ promotes a uniform cluster distribution. By dividing element-wise with the cluster probability, it is effectively up-weighing the summand corresponding to

Figure 1. Visual comparison between C-EDM (C=400) and unconditional EDM (Uncond.) at $M_{img} = 100M$ on FFHQ at 128x128.

classes with low probability. In other words, when $\tilde{q}_t^i(c)$ is low. The hyperparameter γ reduces the influence of the numerator, which leads to partial collapse [5] when $\gamma = 1$.

C.2. What about the lower bound? TEMI with $\gamma = 1$ experiments.

Starting with a high overestimation of the number of clusters (e.g. $1K$ for CIFAR10), we find that TEMI clustering with $\gamma = 1$ utilizes a subset of clusters, which could be used as a lower cluster bound. More precisely, we find a maximum standard deviation of 6.4 for C^u across datasets and feature extractors (see Supp.). Intuitively, C^u is the minimum amount of clusters TEMI (with $\gamma = 1$) uses to group all image pairs. This behavior is analogous to cluster-based self-supervised learning (using image augmentations) [2, 15] and has been recently coined as partial prototype collapse [5]. Nonetheless, the lower bound is more applicable to large scales as the measured standard deviation might exclude the optimal granularity for small, highly curated datasets. Due to the above limitation, we leave this for future work.

As depicted in Tab. 1, the utilized number of clusters C^u is not sensitive to the pre-determined number of clusters nor the choice of backbone for TEMI clustering when $\gamma = 1$.

C.3. TEMI with different backbones.

Here, we report ANMI across various cluster sizes based on the result reported in the main paper (Fig. 5, main paper). For all the conducted experiments, we used TEMI with $\gamma = 0.6$. Apart from having roughly the same FID, we can observe the ranking of backbones w.r.t ANMI is not consistent across cluster sizes.

C.4. Dependence on $q(c)$ during generative sampling on balanced classification datasets.

It is well-established in the clustering literature that k-means clusters are highly imbalanced [14]. To illustrate this in a generative context, we sample from a uniform clus-

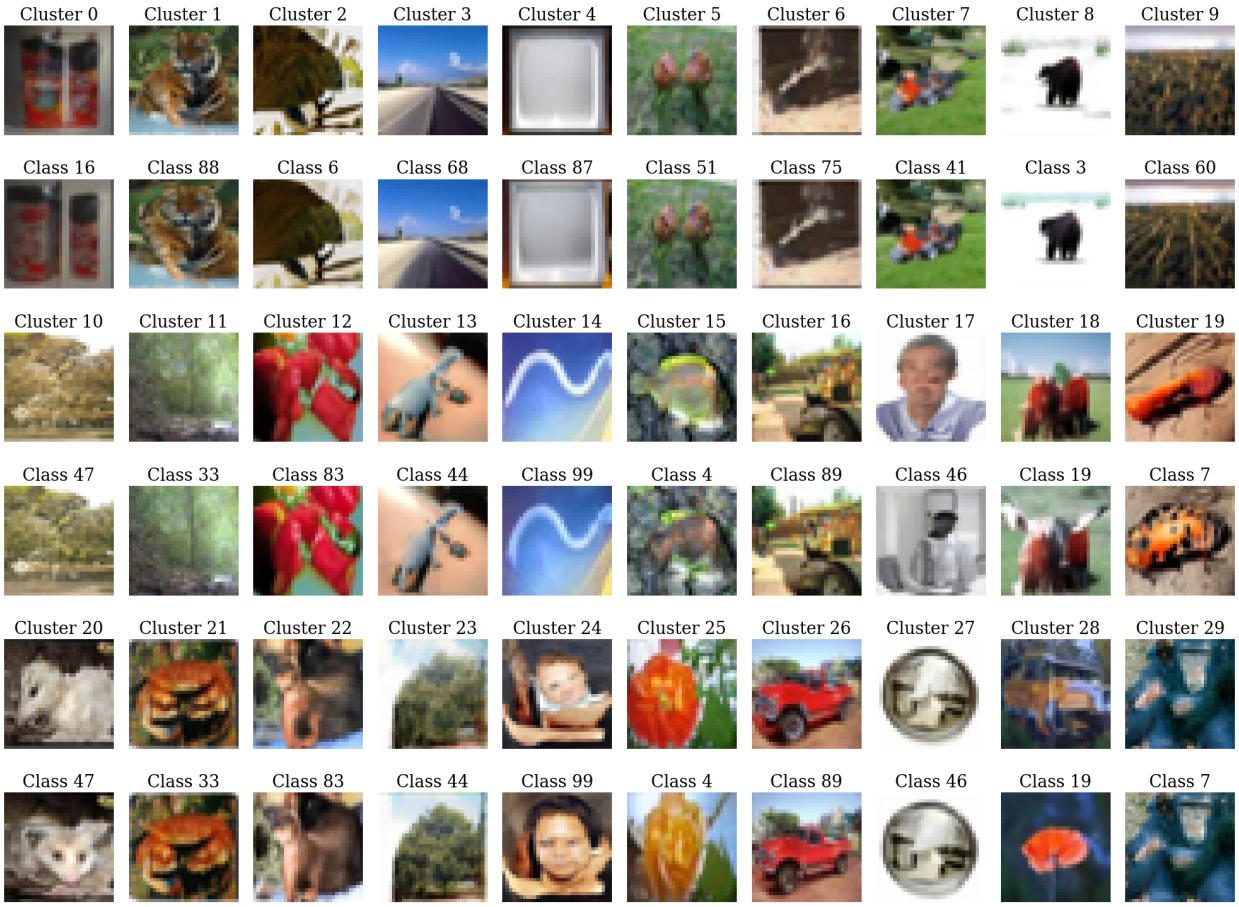


Figure 2. Visualizing generated images from CIFAR100 using C-EDM (even rows) and ground truth conditional EDM (odd rows) using the same initial noise and deterministic noise sampling. We map the C=100 CIFAR100 cluster to the respective ground truth class as computed via the Hungarian one-to-one mapping.

Table 1. Number of utilised clusters C^u for different number of input clusters C (*left*) and different backbones (*right*) using TEMI with $\beta = 1$ with the DINO ViT-B/16 backbone. We show the relatively small sensitivity of C^u to the choice of C and backbone; we report a standard deviation of a maximum value of 6.37 across different cluster sizes and 6.44 across backbones on CIFAR10.

TEMI $\gamma = 1$	CIFAR10 C^u	FFHQ C^u	CIFAR100 C^u
100	33	36	48
400	38	48	48
500	34	54	51
800	40	45	47
1K	34	49	47
2K	48	52	54
5K	28	42	51
Mean	36.4	46.6	49.4
Std	**6.37**	6.16	2.63

TEMI $\gamma = 1, C = 500$	CIFAR10 C^u
DINO ViT-B/16 [2]	34
MoCOv3 ViT-B/16 [3]	39
iBOT ViT-L/14 [15]	45
OpenCLIP ViT-G/14 [4]	47
DINOv2 ViT-g/14 [11]	50
Mean	43
Std	6.44

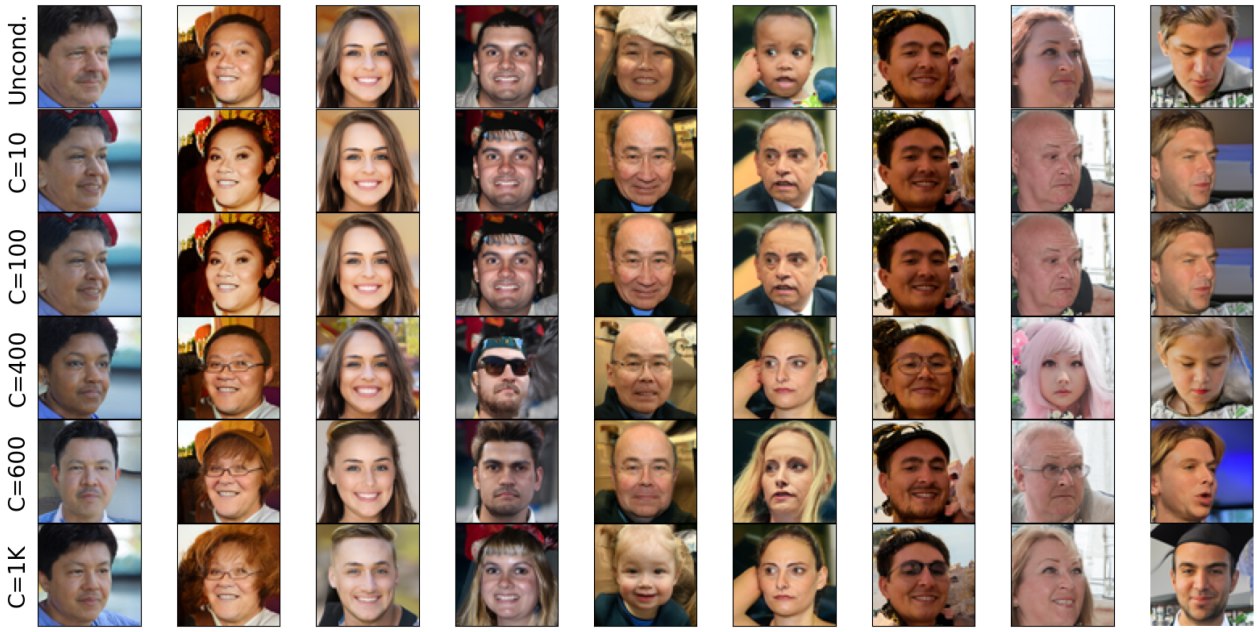


Figure 3. Visualizing *generated* images from FFHQ-64 using CEDM for different number of clusters C with the same random noise. We use deterministic noise sampling. Each noise gets a condition sampled from $p(c)$ for each individual clusters.

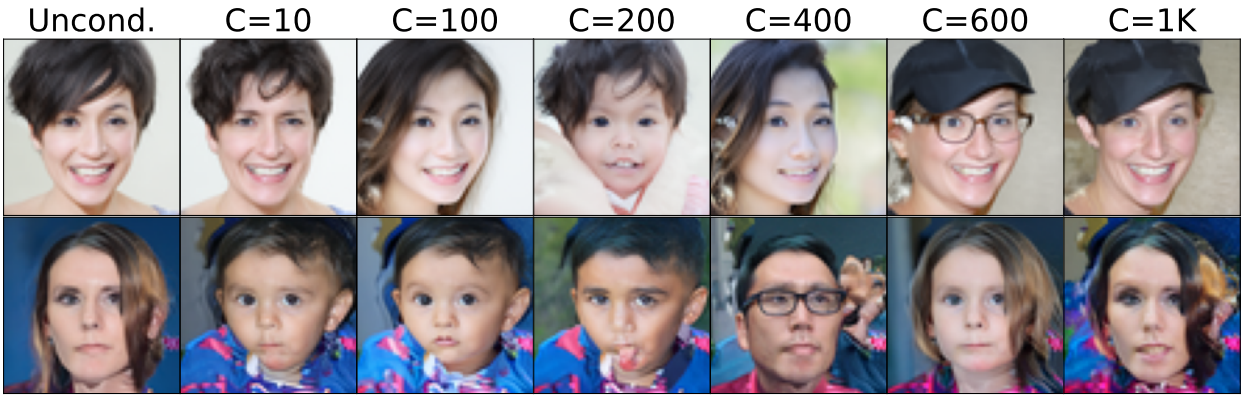


Figure 4. Generated FFHQ-64 samples using *C-EDM* and TEMI clusters with different granularity levels C as well as unconditional EDM (*Uncond.*, first column). All samples in a row use the same initial noise. The cluster assignment is randomly sampled from $q(c)$ for each C.

ter distribution instead of $q(c)$ for balanced classification datasets (CIFAR10 and CIFAR100). As expected, k-means is more dependent to $q(c)$ compared to TEMI, as its FID is significantly deteriorated.

D. The EDM diffusion baseline.

This section briefly summarizes the EDM framework for diffusion models, which was used extensively in this work. For more details and the official EDM code, we refer the reader to the original paper by Karras et al. [7].

Given a data distribution $p_{\text{data}}(\mathbf{x})$, consider the conditional distribution $p(\mathbf{x};\sigma)$ of data samples noised with i.i.d. Gaussian noise of variance σ^2. Diffusion-based generative models learn to follow trajectories that connect noisy samples $\mathbf{x} \sim p(\mathbf{x};\sigma)$ with data points $\mathbf{y} \sim p_{\text{data}}(\mathbf{x})$. Song et al. [12] introduced the idea of formulating the forward trajectories (from data to noise) using stochastic differential equations (SDE) that evolve samples $\mathbf{x}(\sigma)$ according to $p(\mathbf{x};\sigma)$ with $\sigma = \sigma(t)$ as a function of time t. They also

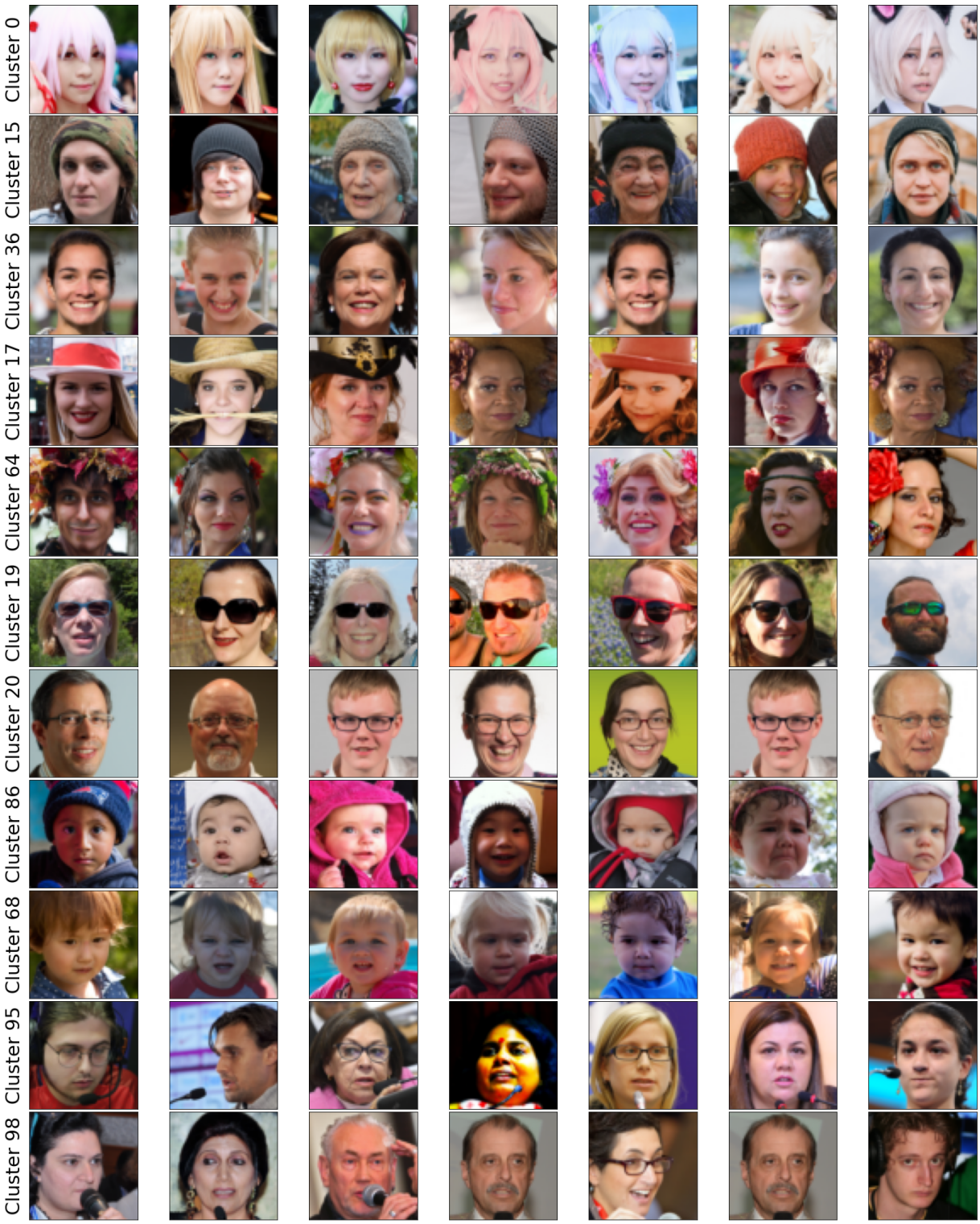
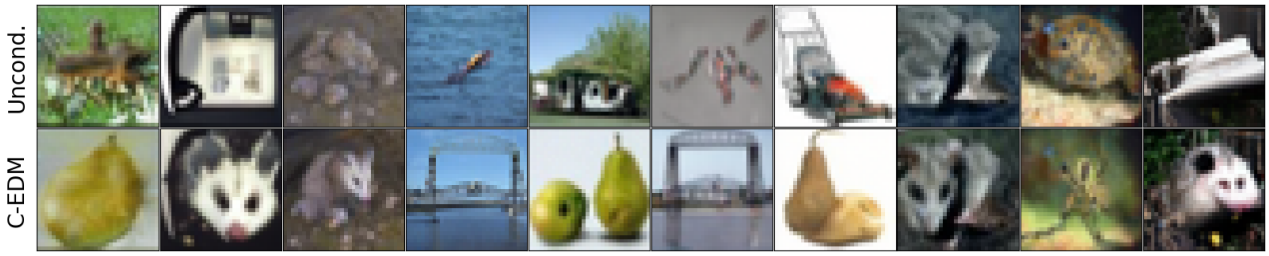


Figure 5. Visualizing *training* images from FFHQ that belong to the same TEMI cluster. Images that are grouped into the same cluster are shown in the same row. We use the trained TEMI model with $C_V = 400$ using the DINO backbone. Cluster assignments are picked to illustrate that images with similar visual characteristics are grouped together (i.e., beanies, smiling faces, glasses, hats, kids, etc.). Images are **randomly** sampled from each cluster.



(a) High-confident C-EDM samples

(b) Low-confident C-EDM samples

Figure 6. Generated low- (*a*) and high-confident (*b*) **CIFAR100** samples . The top row depicts the unconditional (*Uncond.*) samples, while the bottom row shows the generated samples using *C-EDM* with TEMI ($C = 200$). Images on the same column are produced with the same initial noise. Confidence is quantified using maximum softmax probability (MSP). MSP is measured using TEMI trained on CIFAR100 without annotated data.

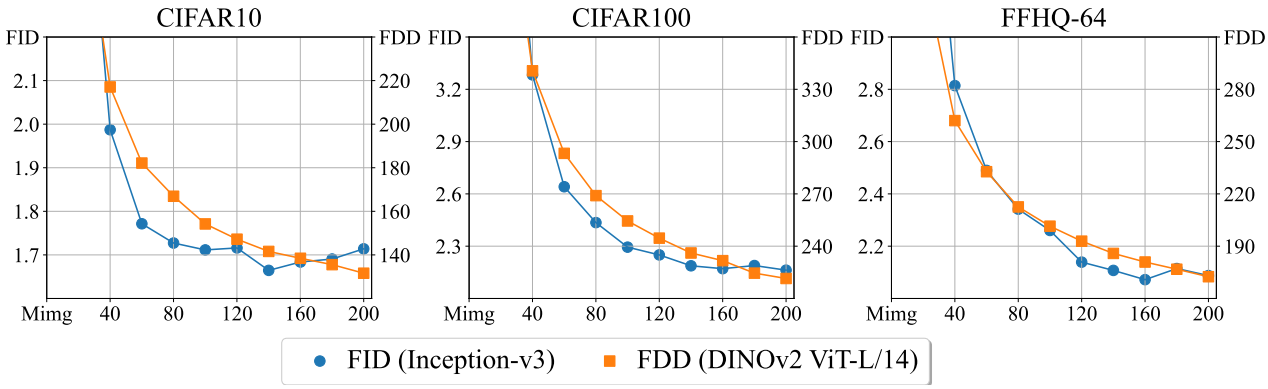


Figure 7. FID score (*y-axis, left*) and FDD (*y-axis, right*) during training samples seen (M_{img}, *x-axis*). We used $C_V = 100, 200, 400$ for CIFAR10, CIFAR100 and FFHQ-64 respectively.

proposed a corresponding "probability flow" ordinary differential equation (ODE), which is fully deterministic and maps the data distribution $p_{\text{data}}(\mathbf{x})$ to the same noise distribution $p(\mathbf{x}; \sigma(t))$ as the SDE, for a given time t. The ODE continuously adds or removes noise as the sample evolves through time. To formulate the ODE in its simplest form, we need to set a noise schedule $\sigma(t)$ and obtain the *score function* $\nabla_{\mathbf{x}} \log p(\mathbf{x}; \sigma)$:

$$\mathrm{d}\mathbf{x} = -\dot{\sigma}(t)\sigma(t)\nabla_{\mathbf{x}} \log p(\mathbf{x}; \sigma)\mathrm{d}t. \tag{1}$$

While mathematical motivations exist for the choice of schedule $\sigma(t)$, empirically motivated choices were shown to be superior [7]. The main component here, the score function, is learned by a neural network through what is known as *denoising score matching*. The core observation here is that the score does not depend on the intractable normalization constant of $p(\mathbf{x}, \sigma(t))$, which is the reason that diffusion models in their current formulation work at all (maybe remove this side-note). Given a denoiser $D(\mathbf{x}, \sigma)$ and the L2-denoising error

$$\mathbb{E}_{\mathbf{y}\sim p_{\text{data}}}\mathbb{E}_{\mathbf{n}\sim\mathcal{N}(0,\sigma^2 I)}[\|D(\mathbf{y}+\mathbf{n}, \sigma) - \mathbf{y}\|^2], \tag{2}$$

we can recover the score function via $\nabla_{\mathbf{x}} \log p(\mathbf{x}, \sigma) = (D(\mathbf{x}, \sigma) - \mathbf{x})/\sigma^2$. Thus, parametrizing the denoiser as a neural network and training it on Eq. (2) allows us to learn the score function needed for Eq. (1). To solve the

Table 2. CIFAR10 ANMI across different cluster sizes and state-of-the-art feature extractors used for TEMI clustering with $\gamma = 0.6$. We only reported the ANMI for $C = 100$ in the main manuscript.

TEMI ($\gamma = 0.6$)	ANMI	ANMI	ANMI
C	50	100	200
MoCov3 ViT-B/16 [3]	65.2	59.3	55.0
DINO ViT-B/16 [2]	65.7	60.7	55.8
DINOv2 ViT-g/14 [11]	66.1	63.8	**59.3**
iBOT ViT-L/14 [15]	68.7	62.7	57.4
CLIP ViT-G/14 [4]	**70.6**	**64.7**	58.9

ODE in Eq. (1), we can put the recovered score function into Eq. (1) and apply numerical ODE solvers, like Euler's method or Heun's method [1]. The ODE is discretized into a finite number of sampling times $t_0, ..., t_N$ and then solved through iteratively computing the score and taking a step with an ODE solver.

Table 3. We report FID for k-means and TEMI with and without considering the training data's cluster distribution $q(c)$. $\mathcal{U}(\{1,..,C\})$ denotes the uniform cluster distribution. We use $C_V = 100, 200, 400$ for CIFAR10, CIFAR100 and FFHQ, respectively. Δ quantifies the absolute difference.

EDM [7]	Sampling Distribution	CIFAR10	CIFAR100
k-means	$\mathcal{U}(\{1,..,C\})$	2.75	2.60
k-means	$q(c)$	1.69	2.21
Δ ($\downarrow$)	-	0.79	0.39
TEMI	$\mathcal{U}(\{1,..,C\})$	1.86	2.41
TEMI	$q(c)$	1.67	2.17
Δ ($\downarrow$)	-	**0.19**	**0.24**

E. Additional implementation details and hyperparameters

When searching for C_V, we evaluate EDM after training with $M_{img} = 100$ and for $M_{img} = 200$ once C_V is found. We only report k-means cluster conditioning with $k = C_V$. All our reported FID and FDD values are averages over 3 runs of 50k images each, each with different random seeds. Below, we show the hyperparameters we used for all datasets to enable reproducibility. We always use the average FID and FDD for three sets of 50K generated images. The used hyperparameters can be found in Tabs. 4 and 5

To assign the CLIP pseudo-labels (Sec. 4.4) to the training set, we compute the cosine similarity of the image and label embeddings using openclip's ViT-G/14 [6]. The label embeddings use prompt ensembling and use the five prompts: a photo of a <label>, a blurry photo of a <label>, a photo of many <label>, a photo of the large <label>, and a photo of the small <label> as in [10].

Table 4. Hyperparameters used for training EDM and C-EDM. **Bold** signifies that the value is changing across datasets. All other parameters of the training setup were identical to the specifications of Karras et. al [7], which are detailed there.

Hyperparameter	CIFAR10/CIFAR100	FFHQ-64/AFHQ-64
Optimization		
optimizer	Adam	Adam
learning rate	0.001	0.001
betas	0.9, 0.999	0.9, 0.999
batch size	1024	512
FP16	true	true
SongUNet		
model channels	128	128
channel multiplier	2-2-2	1-2-2-2
dropout	13%	5% / 25%
Augmentation		
augment dim	9	9
probability	12%	15%

Table 5. TEMI hyperparameters

Hyperparameter	Value
Head hyperparameters	
MLP hidden layers	2
hidden dim	512
bottleneck dim	256
Head final gelu	false
Number of heads (H)	50
Loss	TEMI
γ	0.6
Momentum λ	0.996
Use batch normalization	false
Dropout	0.0
Temperature	0.1
Nearest neibohrs (NN)	50
Norm last layer	false
Optimization	
FP16 (mixed precision)	false
Weight decay	0.0001
Clip grad	0
Batch size	512
Epochs	200
Learning rate	0.0001
Optimizer	AdamW
Drop path rate	0.1
Image size	224

References

[1] Ascher, U.M., Petzold, L.R.: Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations. Society for Industrial and Applied Mathematics, USA, 1st edn. (1998) 7

[2] Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9650–9660 (2021) 2, 3, 7

[3] Chen, X., Xie, S., He, K.: An empirical study of training self-supervised vision transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9640–9649 (2021) 3, 7

[4] Cherti, M., Beaumont, R., Wightman, R., Wortsman, M., Ilharco, G., Gordon, C., Schuhmann, C., Schmidt, L., Jitsev, J.: Reproducible scaling laws for contrastive language-image learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2818–2829 (2023) 3, 7

[5] Govindarajan, H., Sidén, P., Roll, J., Lindsten, F.: On partial prototype collapse in clustering-based self-supervised learning (2024), https://openreview.net/forum?id=Z2dVrgLpsF 2

[6] Ilharco, G., Wortsman, M., Wightman, R., Gordon, C., Carlini, N., Taori, R., Dave, A., Shankar, V., Namkoong, H., Miller, J., Hajishirzi, H., Farhadi, A., Schmidt, L.: Openclip (Jul 2021). https://doi.org/10.5281/zenodo.5143773, https://doi.org/10.5281/zenodo.5143773, if you use this software, please cite it as below. 9

[7] Karras, T., Aittala, M., Aila, T., Laine, S.: Elucidating the design space of diffusion-based generative models. Advances in Neural Information Processing Systems **35**, 26565–26577 (2022) 4, 6, 8, 10

[8] Karras, T., Aittala, M., Lehtinen, J., Hellsten, J., Aila, T., Laine, S.: Analyzing and improving the training dynamics of diffusion models (2023) 1

[9] Kuhn, H.W.: The hungarian method for the assignment problem. Naval research logistics quarterly **2**(1-2), 83–97 (1955) 1

[10] Ming, Y., Cai, Z., Gu, J., Sun, Y., Li, W., Li, Y.: Delving into out-of-distribution detection with vision-language representations. Advances in Neural Information Processing Systems **35**, 35087–35102 (2022) 9

[11] Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023) 3, 7

[12] Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. CoRR **abs/2011.13456** (2020), https://arxiv.org/abs/2011.13456 4

[13] Stein, G., Cresswell, J.C., Hosseinzadeh, R., Sui, Y., Ross, B.L., Villecroze, V., Liu, Z., Caterini, A.L., Taylor, E., Loaiza-Ganem, G.: Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. In: Thirty-seventh Conference on Neural Information Processing Systems (2023) 1

[14] Van Gansbeke, W., Vandenhende, S., Georgoulis, S., Proesmans, M., Van Gool, L.: Scan: Learning to classify images without labels. In: European conference on computer vision. pp. 268–285. Springer (2020) 2

[15] Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., Kong, T.: ibot: Image bert pre-training with online tokenizer. arXiv preprint arXiv:2111.07832 (2021) 2, 3, 7