Research report - MOKAPLAN team

INRIA-CNRS-Paris Dauphine University PSL

Memorization vs. Generalization in diffusion models with the U-Net architecture

A Reproduction with Perceptual Metrics and Attribute-Controlled Splits

Intern
Sylvain Topeza, ENSAE Paris

Supervisors
Vincent Duval, Inria
Antonin Chambolle, CNRS

September 9, 2025

Contents

1 Introduction 2
1.1 Motivation: a very specific generalization 2
1.2 Objectives and summary of this reproduction study 3

2 Reference work and data 4
2.1 Memorization vs Generalization in diffusion models 4
2.2 CelebA dataset: structure and implications for disjoint splits 4
2.3 Perceptual similarity metrics: LPIPS and beyond 5

3 Data and reproducible pipeline 5
3.1 CelebA dataset and metadata 5
3.2 Dataset construction and deterministic ordering 6
3.3 Memory-efficient duplicate removal 6
3.4 Split construction and experimental design 7

4 Models and training 8
4.1 Architecture and objective 8
4.2 Noise schedule and optimization 8
4.3 Training protocol and exposure control 9
4.4 Reproducibility and monitoring 9

5 Experimental results 9
5.1 Baseline reproduction on CelebA 9
5.2 Attribute-controlled splits: Eyeglasses and Male/Female 11
5.3 Identity-disjoint splits 14
5.4 Perceptual re-ranking of nearest neighbors with LPIPS 15

6 Discussion and perspectives 16
6.1 Reassessing the generalization claim 16
6.2 The role of dataset structure 16
6.3 Metric validity and methodological implications 16
6.4 Limitations 17

7 Conclusion 17

8 Appendix 20

1 Introduction

1.1 Motivation: a very specific generalization

Deep generative modeling with score-based diffusion has achieved state-of-the-art sample quality across vision domains, yet the mechanisms underlying generalization remain actively debated. A recent study by Kadkhodaie et al. (2024) reports a striking phenomenon: two denoising networks trained independently on large, non-overlapping subsets of CelebA learn nearly the same score function and consequently generate almost the same images when seeded with identical noise inputs. This observation is framed as evidence for strong generalization arising from inductive biases in the architecture and training that align with geometry-adaptive harmonic representations learned by denoisers. At face value, this suggests that diffusion models may converge to a unique density even with finite and disjoint training data, challenging concerns that generation quality is predominantly driven by memorization of training samples.

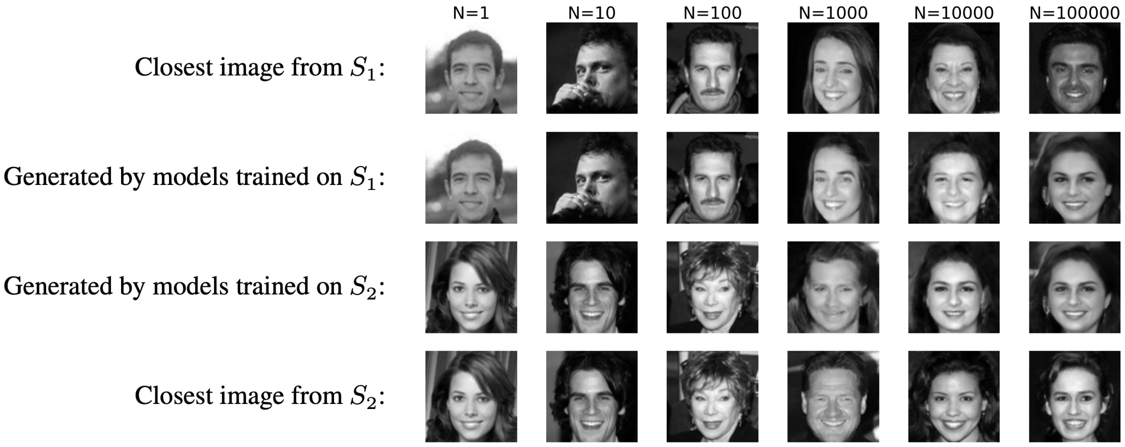

N=1
N=10
N=100
N=1000
N=10000
N=100000
Closest image from S_1:
Generated by models trained on S_1:
Generated by models trained on S_2:
Closest image from S_2:

Figure 1 – Kadkhodaie, Guth, Simoncelli, and Mallat (2024) : two denoisers trained on sufficiently large non-overlapping sets converge to essentially the same denoising function

Despite its appeal, this conclusion is surprising for several reasons tied to both the learning process and the dataset. First, training is partially stochastic, and independent optimization trajectories on disjoint data would not obviously converge to the same function without a powerful shared bias. Second, CelebA includes 202,599 images with 10,177 identity labels and 40 binary attributes, which implies multiple images per identity and a highly structured distribution. This structure creates the possibility that "disjoint" splits by image are nonetheless similar at the level of identities and attributes, potentially making the two training subsets less different than they appear. Third, the evidence for the "originality" of generated samples relative to the training set relies on pixel-wise correlation to identify nearest neighbors; however, correlation is not perceptually aligned and is sensitive to small spatial shifts, making it an unreliable proxy for semantic proximity in images. Perceptual metrics better reflect human similarity judgments.

1.2 Objectives and summary of this reproduction study

This report presents a student-led research replication and methodological contribution conducted within the Mokaplan team at Inria. The project faithfully reproduces the core experiments on CelebA using the authors' public code and protocol while operating under constrained compute, and then extends the analysis to address two issues: metric validity for train-nearest-neighbor assessment and the effective heterogeneity of the training splits. Concretely, the work:

- Reconstructs the CelebA training tensor at 40×40 resolution rather than the 80×80 used in the original study, due to computational constraints, while confirming that the headline phenomenon ("same noise → similar images across independently trained denoisers") persists qualitatively at 40×40. The reconstruction uses the authors' preprocessing logic and preserves a deterministic image order, recorded to disk, to guarantee reproducible disjoint splits.
- Implements a memory-efficient de-duplication procedure to eliminate highly correlated near-duplicate images from the dataset tensor, replacing the repository's dense all-pairs correlation approach—which is impractical at CelebA scale—with a blockwise computation that identifies the same duplicates but with dramatically reduced RAM requirements. This modification is critical for feasibility on typical research clusters and maintains parity with the original duplicate-removal criterion.
- Challenges the use of pixel correlation for nearest-neighbor comparisons by introducing LPIPS (VGG backbone) into the evaluation pipeline, following recommended input normalization and three-channel expansion, and demonstrates substantial rank inversions between correlation and LPIPS on top-k candidates, consistent with LPIPS's stronger alignment to human perception. This re-ranking indicates that the "closest training image" reported by correlation is often not the perceptually closest instance, which weakens claims of sample originality.
- Designs and initiates attribute-controlled experiments that increase the heterogeneity between disjoint training subsets by splitting CelebA using its official attribute annotations, beginning with *Eyeglasses* and extending to *Male/Female*. These experiments probe whether cross-model alignment under identical noise persists when training distributions diverge more significantly along semantic axes.
- Constructs *identity-disjoint* splits using CelebA's identity metadata so that each celebrity appears in exactly one group. Retraining under this constraint tests whether identity overlap stabilizes the "same-noise → same-image" convergence, or whether convergence persists in the absence of any cross-identity redundancy.

In subsequent sections, we detail the reference work, data pipeline, training protocol, evaluation design, and preliminary observations.

2 Reference work and data

2.1 Memorization vs Generalization in diffusion models

The memorization–generalization trade-off in generative modeling has a long history. Early GANs often suffered from mode collapse and overfitting, producing outputs that lacked diversity or even memorized training samples (Arora and Zhang, 2017). In contrast, diffusion models have recently achieved much stronger data coverage and diversity, suggesting improved generalization (Ho et al., 2020).

For diffusion, the training objective can be interpreted as learning score functions $\nabla_x \log p_\sigma(x)$ across different noise levels, which in principle encourages learning the underlying data manifold rather than memorizing individual samples. However, recent evidence shows that large generative models, including diffusion models, can still memorize or partially memorize training data under certain conditions—such as limited dataset diversity, the presence of near-duplicate samples, or repeated exposure during training (Somepalli et al., 2022). Such findings highlight vulnerabilities to replication and raise concerns that sample originality may be overstated.

In contrast, Kadkhodaie et al. (2024) report a different phenomenon by demonstrating that two UNet denoisers trained independently on large, non-overlapping subsets of CelebA converge to nearly identical score functions and generate strikingly similar faces when seeded with the *same* noise. This behavior suggests a convergence toward geometry-adaptive harmonic representations driven by architectural inductive biases. While this points to a stronger form of generalization than previously assumed, this apparent contradiction may partly reflect differences in data regimes, but it nonetheless underscores the need for careful methodological choices—both in dataset design and in evaluation metrics—when assessing whether diffusion models truly generalize beyond their training data.

2.2 CelebA dataset: structure and implications for disjoint splits

The CelebFaces Attributes Dataset or CelebA (Liu et al., 2015) contains 202,599 aligned and cropped celebrity face images spanning 10,177 unique identities, with each image annotated for 40 binary attributes including Eyeglasses, Male, Smiling, Young, and others. The dataset's rich structure has important implications for experiments involving "disjoint" training splits. With an average of approximately 20 images per identity, random splits of CelebA have a high probability of including multiple images of the same person across different subsets, creating latent identity overlap even when image sets are formally disjoint.

This structure is particularly relevant for generative modeling studies, as it creates the possibility that apparent "generalization" between models trained on different subsets actually reflects learning from multiple perspectives of the same individuals rather than genuine cross-identity generalization. The attribute annotations provide an opportunity to create more controlled splits: for example, the Eyeglasses attribute divides the dataset into 13,193 images with glasses and 189,406 without, while the Male attribute creates roughly balanced gender-based partitions (84,434 vs. 118,165).

2.3 Perceptual similarity metrics: LPIPS and beyond

Traditional image similarity metrics such as Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index (SSIM) often fail to capture perceptual similarity as judged by humans, particularly for natural images where semantic content may be preserved despite pixel-level differences. The Learned Perceptual Image Patch Similarity (LPIPS) metric, introduced by Zhang et al. (2018), addresses this limitation by computing similarity between deep feature activations from pre-trained networks such as AlexNet, VGG, or SqueezeNet. LPIPS has been extensively validated against human perceptual judgments and has become a standard evaluation tool in computer vision and graphics applications.

Recent extensions to LPIPS have addressed specific limitations, including shift-tolerance for small spatial misalignments and adversarial robustness (Ghildyal and Liu, 2022). The shift-tolerant variant is particularly relevant for face image analysis, where small translations or rotations should not dramatically affect perceptual similarity. For implementation, LPIPS requires careful input preprocessing: images must be normalized to [-1,1] range and provided as three-channel tensors, even for grayscale inputs, to ensure compatibility with pre-trained backbone networks.

The choice of backbone network (AlexNet vs VGG vs SqueezeNet) can influence LPIPS scores, with VGG generally providing more stable and interpretable results for face images due to its deeper feature representations. The metric produces lower scores for more perceptually similar image pairs, making it suitable for nearest-neighbor ranking tasks where the goal is to identify the most semantically similar training examples to a generated sample.

3 Data and reproducible pipeline

3.1 CelebA dataset and metadata

For this reproduction, we utilized almost the complete CelebA dataset, as our objective was to maximize the available training data for each model while maintaining strict disjointness between splits. Of the 202,599 total images, we set aside 10 that are not used in our main experiments. In principle they could serve as a tiny common test pool for both groups, but in practice our evaluation relies on the cross-train/test protocol described later (what is train for one model serves as test for the other).

All attribute annotations were preserved to support controlled split experiments and heterogeneity testing. However, the attribute file provided with the CelebA distribution contained formatting errors—specifically, occasional line breaks that split an image entry across two lines—making it unreadable in Python without preprocessing. As a preliminary step, we repaired this file by detecting and correcting misaligned rows, ensuring that every image was associated with a consistent vector of 40 binary attributes. This correction was crucial for enabling attribute-based partitioning in later experiments and for guaranteeing the integrity of metadata throughout the pipeline.

3.2 Dataset construction and deterministic ordering

A critical reproducibility gap in the original study was the lack of deterministic image ordering, which makes it impossible to verify that "disjoint" splits are identical across different reproductions. To address this, we implemented a fully deterministic preprocessing pipeline that records the exact order of images in the final .pt tensor.

The preprocessing pipeline follows the authors' `celebA_to_torch.py` script with several key modifications for reproducibility and resource constraints:

```
all_ims = load_CelebA_dataset(
    train_folder_path, test_folder_path, s=0.25,  # for 40x40 resolution
    train_list_path=train_list_path,
    test_list_path=test_list_path)
```

The scaling factor s=0.25 produces 40×40 resolution images instead of the 80×80 used in the original study, representing a necessary compromise for computational constraints while preserving the essential qualitative phenomena. Images are first cropped to 160×160, then downsampled using bicubic interpolation, and finally converted to grayscale.

The order of images in the final tensor is determined by a random ordering of filenames in the source directories and is recorded in a persistent text file (`train_filenames.txt`). This deterministic approach enables full traceability and verification of split construction across different research groups.

3.3 Memory-efficient duplicate removal

The original repository includes a function remove_repeats that removes near-duplicate images by computing the full N×N correlation matrix between all image pairs and removing one image from each pair with correlation above 0.95. However, this approach requires storing a 202,589×202,589 matrix of floating-point values, demanding over 100GB of RAM and proving infeasible on standard research hardware. We developed an alternative procedure, `remove_repeats_block`, that achieves identical duplicate removal while keeping memory usage manageable through blockwise computation (full implementation provided in the Appendix).

The key insight is to compute correlations in blocks of size b and discard each block as soon as high-correlation pairs have been extracted. This method reduces peak memory from $O(N^2)$ floats to $O(b^2)$ (here $b = 5000$), while leaving the decision rule unchanged: mean-removed correlation on 20×20 downsampled images with a threshold of 0.95. We verified empirically that the set of removed indices is identical to the dense baseline, thus maintaining mathematical equivalence with the dense approach.

In our run, the procedure removed exactly 6,354 near-duplicates from 202,589 images, yielding 196,235 unique images (so at most 98,117 per disjoint split). The memory efficiency gain is substantial: whereas the dense baseline would allocate $\approx 202{,}589^2 \times 4$ bytes $\approx 164\,\text{GB}$ for a single float32 matrix, the blockwise variant peaks around $2 \times 5000^2 \times 4 \approx 400\,\text{MB}$ for inputs and temporaries, making the pipeline feasible on standard GPUs/CPUs.

3.4 Split construction and experimental design

Given the cleaned dataset of 196,235 images, we construct disjoint training splits for our two groups of denoisers following the original study's protocol. For each target size $N \in \{10, 100, 1{,}000, 10{,}000, 100{,}000\}$, we define:

— **Group A:** {1, 2, . . . , N},

— **Group B:** {N+1, N+2, . . . , 2N}

This construction ensures strict disjointness by design, as the splits draw from opposite ends of the deterministically ordered tensor. The maximum feasible size is constrained by the cleaned dataset size, with N = 100,000 requiring that N $\leq$ floor(196,235/2) = 98,117 to maintain disjointness. We use N = 98,117 (for readability some figures round this to 100,000) as the largest split size.

For attribute-controlled experiments designed to increase split heterogeneity, we leverage CelebA's binary attribute annotations to create semantically distinct subsets:

— **Eyeglasses-based splits:** using the Eyeglasses attribute, we separate images into approximately 13,000 with glasses and 13,000 without glasses. This creates training subsets that are both image-disjoint and semantically divergent along a salient visual attribute, providing a stronger test of cross-model generalization than random splits.

— **Gender-based splits:** the Male attribute enables balanced splits of roughly 84,000 male and 84,000 female images, creating even larger semantically distinct training sets for testing generalization limits.

These attribute-controlled splits are built as follows. We filter the full dataset by the chosen attribute to obtain the two raw sets. We keep every image on the minority side, and we randomly downsample the majority side to the same size. Each side is then shuffled and its exact filename order is saved to a text file. This file fixes the randomness and guarantees that the splits are exactly reproducible without requiring a specific random seed. Finally, we create a single `.pt` file by concatenating the two lists in that fixed order. This yields two strictly attribute-pure and balanced halves. Group A is trained on the first half and Group B on the second half.

Note on duplicate removal. For these attribute-controlled splits we *do not* apply correlation-based duplicate removal prior to building the subsets. Removing near-duplicates from these subsets, which are already much smaller, would reduce effective diversity and could break exact balance and attribute purity when forming the single concatenated `.pt` file. We therefore keep potential near-duplicates here, with the trade-off of slightly higher redundancy. This choice differs from the baseline and identity-disjoint experiments, where we do remove duplicates as described in Section 3.3.

4 Models and training

4.1 Architecture and objective

All denoisers in this study implement the UNet backbone provided in the original repository, with the sole modification of adapting the input size to 40×40 images rather than 80×80. The network consists of three encoder blocks, one mid-level block, and three decoder blocks. Each block contains two 3×3 convolutions (padding 1) without bias, followed by batch normalization implemented without mean subtraction or offset, and ReLU activations. Encoder blocks apply 2×2 downsampling with a doubling of channels, while decoder blocks apply 2×2 upsampling with channel halving. Skip connections concatenate encoder and decoder feature maps at corresponding resolutions, preserving fine spatial information.

Each model therefore has 7,659,264 parameters. The training protocol follows the standard diffusion score-matching practice with random noise injection at varying levels. At each iteration, a clean image x is sampled from the dataset, a noise level σ is drawn from $[0, 1]$, and Gaussian noise $z \sim \mathcal{N}(0, I)$ is added to form the corrupted observation

$$y = x + \sigma z.$$

The UNet denoiser $f_\theta(y)$ is trained to approximate the conditional mean $\mathbb{E}[\, x \mid y \,]$ by minimizing the mean squared error

$$\mathcal{L}(\theta) = \mathbb{E}_{x,\sigma,z}\left[\, \|x - f_\theta(x + \sigma z)\|_2^2 \,\right],$$

following the denoising–score matching equivalence of Vincent (2011); Raphan and Simoncelli (2011).

Once trained, the denoiser provides an estimate of the score function via the Robbins–Miyasawa identity (Robbins, 1956; Miyasawa, 1961):

$$s_\theta(y) = \frac{f_\theta(y) - y}{\sigma^2} \approx \nabla_y \log p_\sigma(y).$$

This formulation ensures that minimizing reconstruction error directly yields an unbiased estimator of the gradient of the log-density at noise level σ.

4.2 Noise schedule and optimization

Noise levels are sampled independently for each training example. For each image, σ is drawn uniformly from [0, 1], forcing the denoiser to handle the entire spectrum of corruption levels. This choice reproduces the schedule of the original repository.

Training uses the Adam optimizer with learning rate 10^{-3}, decayed by a factor of 0.5 every 100 epochs, up to 1000 epochs. The batch size is 512, and data loading uses two workers. Grayscale images are provided as single-channel tensors.

4.3 Training protocol and exposure control

A key methodological concern is ensuring fair comparison across dataset sizes. Without correction, models trained on small subsets would be underexposed compared to those trained on large subsets. To equalize exposure, each subset of size N is replicated until it reaches 100,000 training samples. For example, with N=100, each image is repeated 1000 times, each time with independent corruptions. This replication scheme isolates the effect of dataset diversity from training exposure. The original study used 250,000 repetitions at 80×80; here we use 100,000 at 40×40 as a computationally feasible compromise that still guarantees convergence. The replication is performed via simple repetition of the base subset, following the `repeat_images` function in the authors' training script.

We train five UNets on group A and five on group B, one for each dataset size $N \in$ {10, 100, 1,000, 10,000, 100,000}, totaling ten models per experiment.

4.4 Reproducibility and monitoring

All experiments are designed for full reproducibility. The dataset tensor is built from the *raw* CelebA image pool, with duplicates removed (in the case of baseline and identity-disjoint experiments) using the blockwise correlation method described in Section 3.3. Before that, the exact order of images is logged to a text file, enabling identical reconstruction of splits across runs and external auditability. Model checkpoints, training losses, and PSNR trajectories are saved at every epoch. Sampling experiments later in the report use identical seeds across groups A and B, allowing side-by-side comparison of denoisers under identical noise conditions.

To ensure transparency and facilitate reuse, all code, preprocessing scripts, and experiment notebooks are released on GitHub [1]. These files are based on the original repository of Kadkhodaie et al. (2024) [2], with only a few modifications introduced for reproducibility and computational feasibility. By contrast, the trained models, preprocessed datasets (lightweight `.pt` tensors), and logs are original contributions of this work, and are hosted on Hugging Face [3].

5 Experimental results

5.1 Baseline reproduction on CelebA

Setup. We follow the original protocol: two groups (A, B) are trained on *disjoint*, deterministically constructed subsets of CelebA (Section 3), for five dataset sizes $N \in \{10, 100, 1{,}000, 10{,}000, 100{,}000\}$. Each subset is replicated to reach 100,000 training samples (Section 4.3). For sampling, we fix seeds across A/B to compare their generations under identical noise.

[1] https://github.com/Sylvain-Topeza/inria-internship

[2] https://github.com/LabForComputationalVision/memorization_generalization_in_diffusion_models

[3] https://huggingface.co/collections/Sylvain-Topeza/inria-internship-68b1c4c4ce489542ef13a2fe

One-shot denoising and PSNR. As in the paper, we probe memorization vs. generalization by denoising images that are *train* for one model and *test* for the other (Figure 2). For small N, the model for which the image is *train* denoises it sharply while the other model tends to regress toward a face from its own training distribution; for large N, both models produce coherent faces even on the other's *train* image, with close visual quality. See the PSNR trend on train vs. test across noise levels:

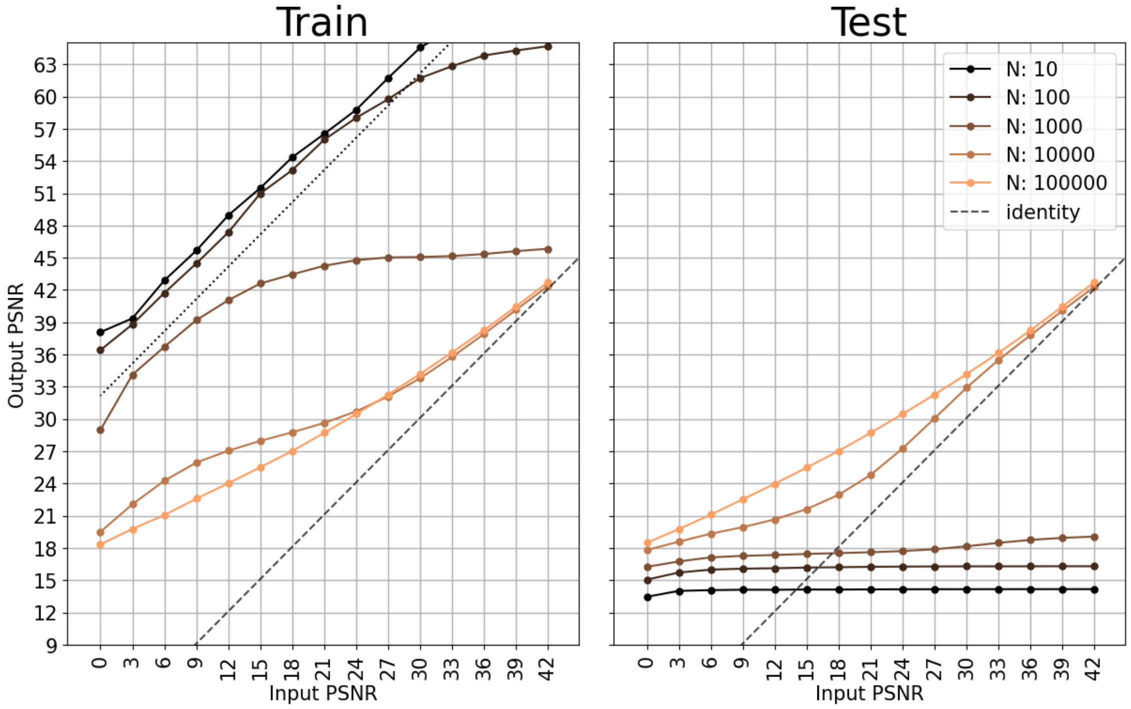


Figure 2 – Output PSNR vs. input PSNR on *train* (left) and *test* (right), for $N \in \{10, 100, 1{,}000, 10{,}000, 100{,}000\}$. The train/test gap shrinks with N.

Same-noise sampling: convergence of A and B. With identical noise seeds, A and B produce noticeably similar faces as N grows (see Figure 3); at $N = 10^5$, samples are often near-indistinguishable up to small details, consistent with the original claim (at 40×40 as in our setup).

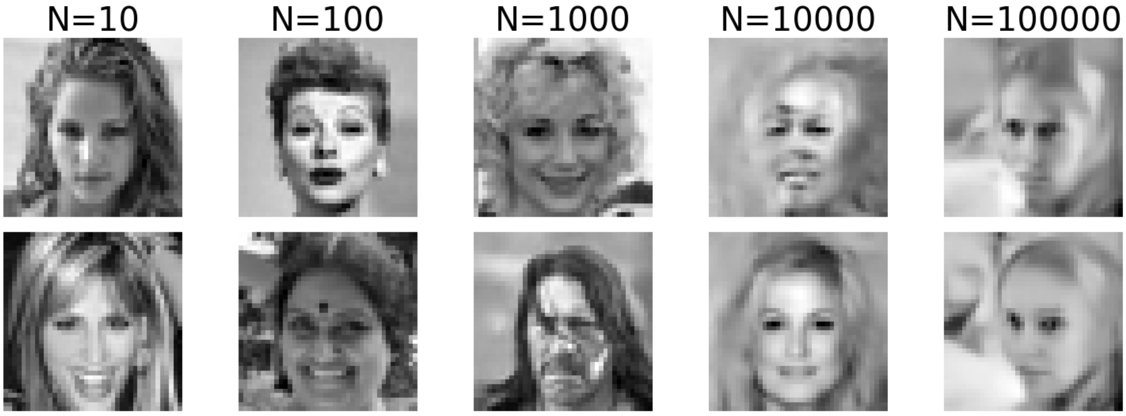


Figure 3 – Same-noise generations for groups A (top) and B (bottom) across N. Qualitative similarity tightens with N.

Closest training examples (correlation-based). Replicating the paper's visualization, Figure 4 shows, for each N, the generated sample and the *closest* training image in A and in B according to pixel correlation. This mirrors the original presentation and will serve as baseline for our LPIPS critique in Section 5.4.

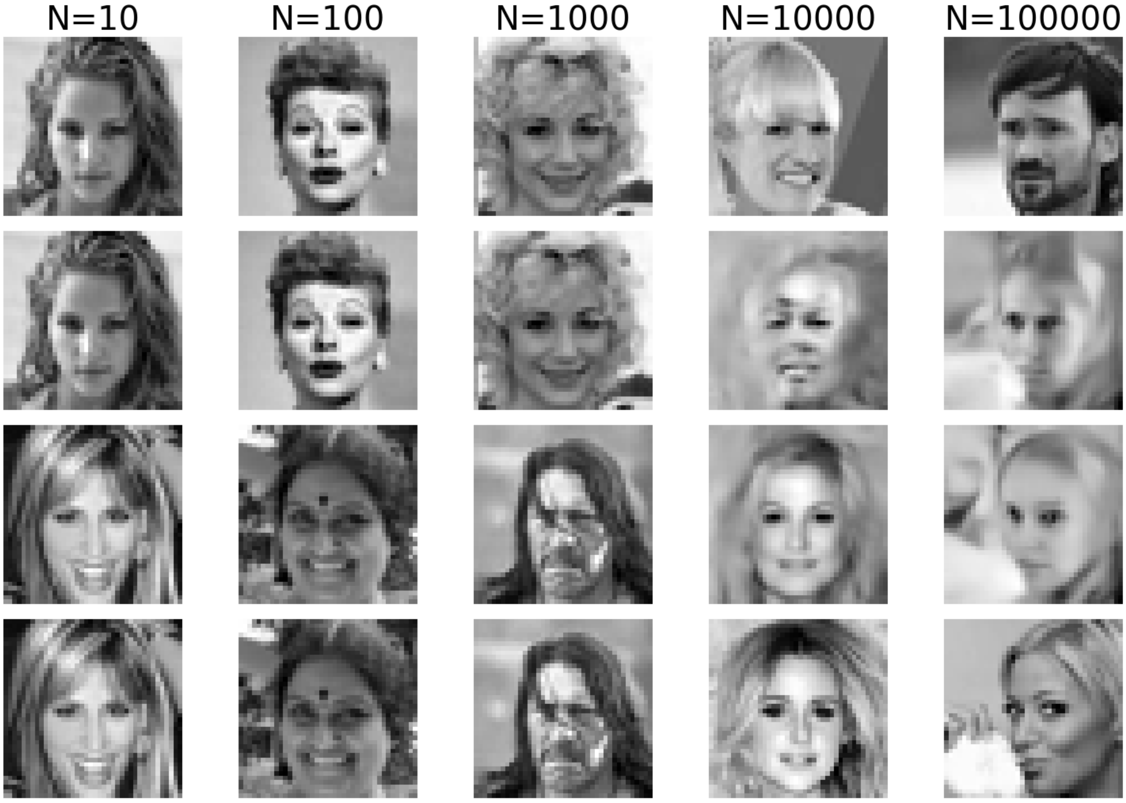


Figure 4 – Generated samples (rows 2–3) and correlation-based *closest* training faces (rows 1, 4) for A/B and each $N \in \{10, 100, 1{,}000, 10{,}000, 100{,}000\}$.

5.2 Attribute-controlled splits: Eyeglasses and Male/Female

Rationale. CelebA's structure (multiple images per identity, 40 attributes) implies that "disjoint by image" does not necessarily imply heterogeneity in identities or attributes. To stress-test cross-model alignment, we construct splits that are disjoint not only by image but also *semantically* different along a single attribute.

Eyeglasses (with vs. without). Using CelebA's `Eyeglasses` annotation, we assemble two balanced subsets of 13,193 images each: *with glasses* and *without glasses*. We train one denoiser on each half (replicated to 100,000 training examples; Section 4.3) and sample with identical seeds. Results presented in Figure 5 are as expected: the "glasses model" generates faces with glasses, whereas the "no-glasses model" generates faces without glasses. At this scale, we do *not* consistently observe face-by-face convergence (same face across A/B) as in random splits with $N = 10^5$; the attribute constraint reduces the effective overlap of distributions and thus weakens the same-noise alignment.

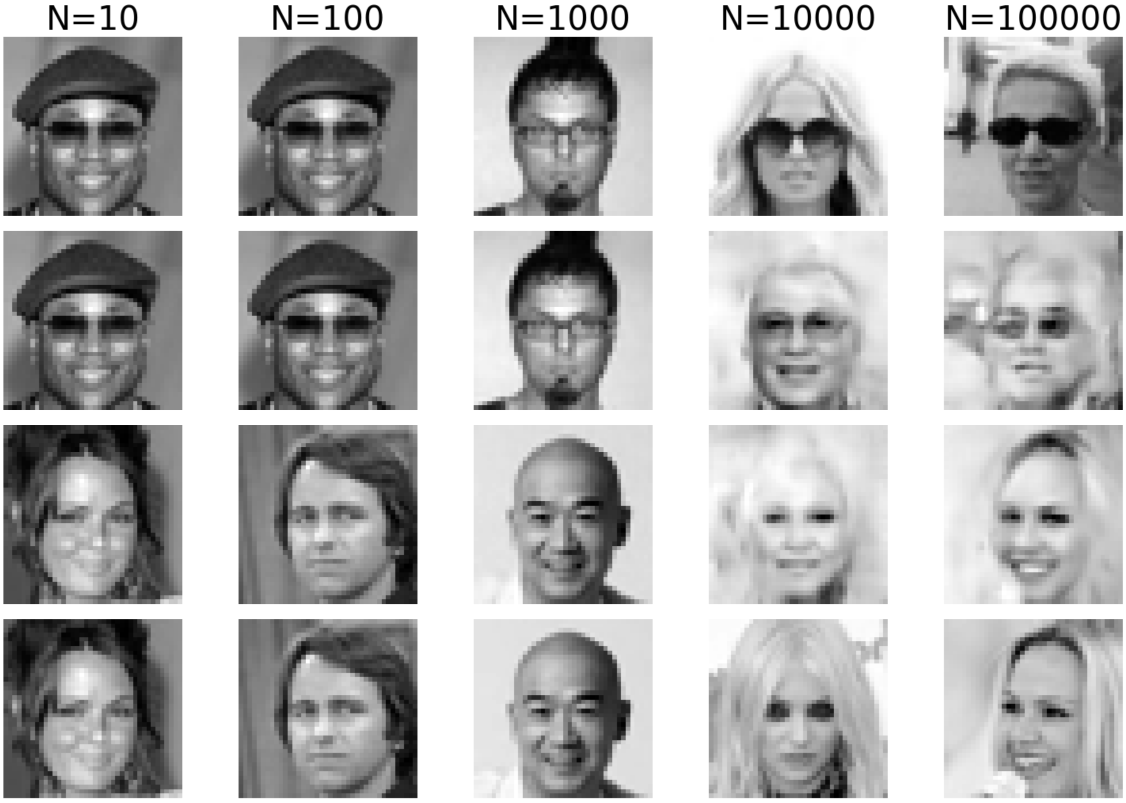


Figure 5 – Attribute-controlled split: *Eyeglasses*, with correlation-based *closest* training faces. Top: denoiser trained on "with glasses" dataset; Bottom: "without glasses". The attribute is respected, convergence is not systematic at this size.

Male vs. Female. Using the `Male` attribute, we split CelebA into two large sets of 84,434 images (male vs. female faces) and train one group of denoisers per set, each replicated to 100,000 training samples. Under identical noise seeds, both models produce a plausible face of the expected gender, confirming that the models remain faithful to the attribute constraint. Beyond this expected divergence in gender, the degree of *cross-model convergence* varies markedly with the random seed:

— **Strong convergence:** in some cases, both denoisers generate almost the same identity, with highly similar facial structure, hair, and pose (see Figure 6). The sole difference is the apparent gender of the face—for example, the "female" model may generate a woman and the "male" model a man with near-identical jawline, hairstyle outline, and gaze direction. This resembles the near-perfect convergence observed in the original random-split experiments (Section 5.1), except with gender as a systematic divergence axis.

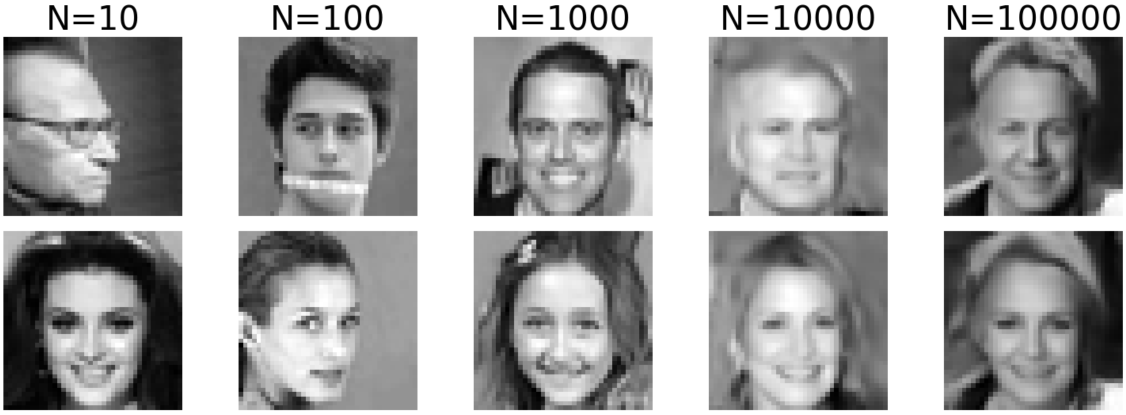


Figure 6 – Example of strong convergence: nearly identical faces apart from gender.

- **Intermediate convergence:** for other seeds, posture and coarse geometry (head orientation, lighting, or facial outline) are aligned across A/B, but fine details diverge significantly (see Figure 7). The "male" and "female" outputs share global structure yet correspond to different perceived individuals.

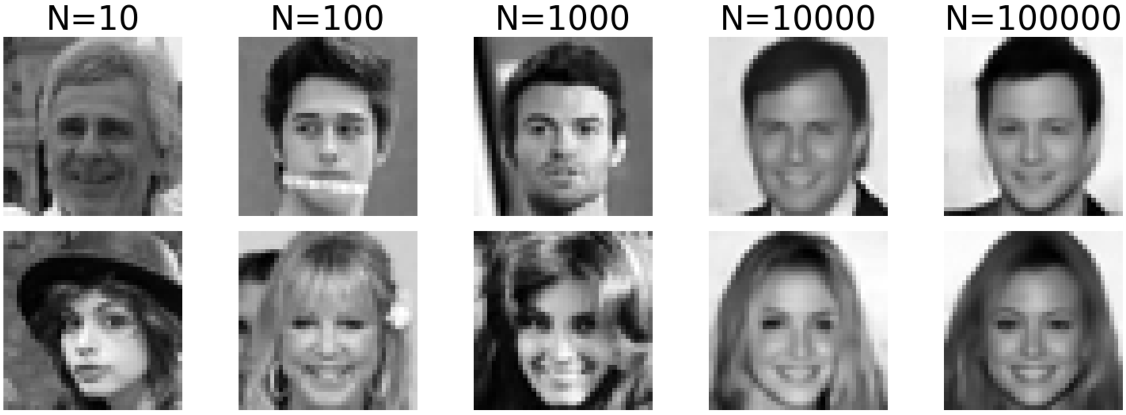


Figure 7 – Example of intermediate convergence: same posture, different identities.

- **Weak convergence:** finally, certain seeds lead to almost no shared structure (see Figure 8). The two outputs differ not only in gender but also in pose, orientation, and even facial proportions, yielding pairs of images with minimal resemblance. These cases illustrate the limits of alignment when the underlying training distributions are more heterogeneous.

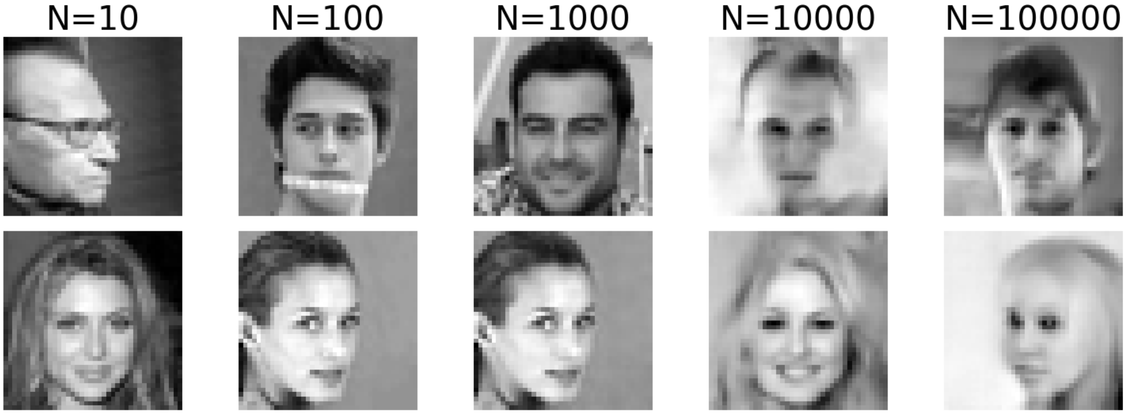


Figure 8 – Example of weak convergence: minimal similarity across outputs.

Overall, this variability highlights that convergence is not binary but exists on a spectrum depending on the seed. Large dataset size ($\approx 10^5$ images) makes strong convergence *possible*, but attribute-driven heterogeneity frequently induces non-trivial divergence. The male/female experiment therefore reveals both the robustness and the limits of inductive biases in U-Net denoisers: models can sometimes map identical noise to semantically aligned but gender-differentiated faces, while in other cases the same noise leads to entirely distinct individuals. Poses and coarse geometry often align across A/B, but identity and fine attributes differ—suggesting partial alignment in low-frequency structure with divergence in identity-specific details.

5.3 Identity-disjoint splits

Rationale. The original "disjoint" protocol (Section 3.4) does not control for identity-level redundancy: the same celebrity may appear in both groups A and B through different photographs. Given that CelebA has on average ~ 20 images per identity, this raises the possibility that part of the strong cross-model convergence is stabilized by overlapping identities. To evaluate this, we constructed identity-disjoint splits in which each celebrity identity is assigned exclusively to one group, ensuring that no individual appears in both training sets.

Quantitative behavior. Models trained on identity-disjoint subsets achieve denoising performance comparable to those trained with overlap. PSNR curves on held-out test images are statistically indistinguishable across the two protocols, indicating that identity-level overlap is not required for denoisers to learn effective score functions. Training dynamics are therefore robust to this constraint at the aggregate level.

Qualitative convergence. Same-noise sampling reveals more differentiated behavior. For many seeds, identity-disjoint denoisers still converge to visually similar outputs, confirming that architectural and training biases are sufficient to drive cross-model alignment. However, for certain seeds the divergence is significantly stronger than under the overlap protocol. Figure 9 illustrates such a case: at $N = 100{,}000$, the overlap-trained models produce nearly identical faces (see Figure 3), while the identity-disjoint models generate two unrelated individuals under the same seed. This effect, absent in the overlap setting, shows that identity redundancy acts as a stabilizer of convergence across seeds.

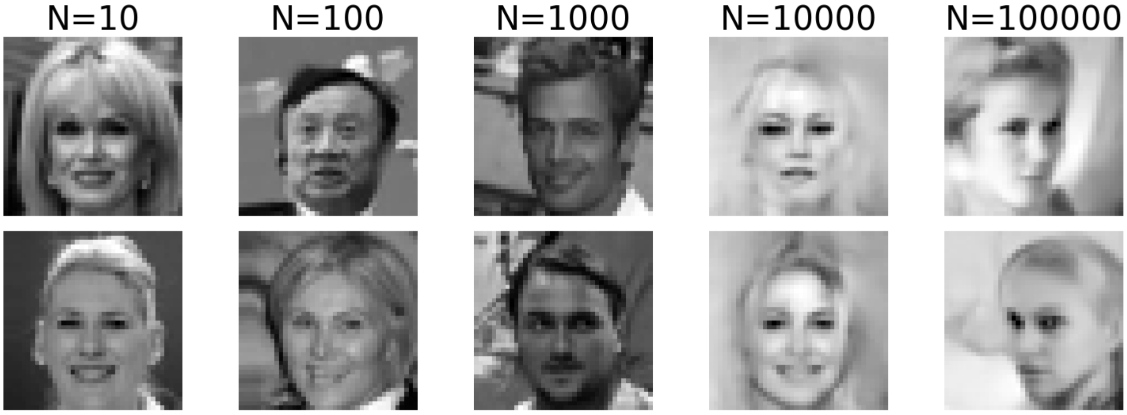


Figure 9 – Same-noise generations at $N = 100{,}000$ under the identity-disjoint protocol as opposed to the overlap protocol highlighted in Figure 3. Identity-disjoint models diverge more strongly under identical seeds.

5.4 Perceptual re-ranking of nearest neighbors with LPIPS

Why correlation is misleading. Pixel correlation is sensitive to small shifts and brightness changes and is poorly aligned with human perception. In Figure 10 (see Appendix), several top-k correlation neighbors for a given generation are plainly dissimilar to the human eye (e.g., mismatched gender, age, or head pose), yet attain similar correlation scores.

Method. For each generated sample, we first retrieve the top-k (here $k = 10$) training images by correlation (as in the original notebook), then we *re-rank* these candidates using LPIPS (VGG backbone) after proper normalization (Section 2.3). This concentrates LPIPS computation on plausible candidates while exposing ordering discrepancies.

Findings. LPIPS re-ranking consistently changes the identity of the *nearest* neighbor among the top-k correlation candidates and reshuffles their order to align with human perception. In many cases, the LPIPS-closest training image is a near-lookalike of the generated sample, while the correlation-closest is not. See Figure 11 (Appendix), where we annotate both the correlation and LPIPS scores for each candidate.

Implication. The "originality" claims based on correlation-only nearest neighbors are over-optimistic: when assessed with a perceptual metric, generated faces are frequently closer to training instances than correlation suggests, meaning that correlation overestimates this originality by overvaluing pixel-level alignment and undervaluing semantic structure. This undermines the narrative that models produce wholly novel identities simply because the correlation-closest example looks different.

6 Discussion and perspectives

6.1 Reassessing the generalization claim

Our reproduction confirms the central phenomenon reported by Kadkhodaie et al. (2024): when trained on large disjoint subsets of CelebA, independently initialized UNet denoisers converge to nearly identical score functions and yield qualitatively indistinguishable samples under identical noise. This provides compelling evidence that inductive biases and optimization dynamics are sufficient to enforce alignment across models, at least in sufficiently redundant data regimes.

However, our extensions show that this convergence should be interpreted with caution. When we introduce attribute-controlled splits (eyeglasses, male/female), cross-model alignment weakens considerably. With glasses vs. no-glasses, the two denoisers reliably respect their attribute constraint, but do not converge to the same underlying identity even at relatively large N. With male vs. female, convergence exists but is inconsistent: certain seeds lead to near-identical individuals differing only in gender, whereas others diverge completely. Similarly, when identity overlap is removed, convergence remains largely possible but becomes less stable across seeds. These results indicate that geometry-adaptive harmonic representations alone cannot fully explain generalization. Dataset redundancy, both in identity and attributes, acts as a significant stabilizer. Generalization in this setting should therefore be interpreted as convergence under *favorable structural conditions*, rather than a universal property of diffusion models.

6.2 The role of dataset structure

CelebA is a highly homogeneous dataset: faces are aligned, cropped, and relatively low in variation compared to natural image corpora. Furthermore, each identity has on average 20 images, ensuring multiple redundant views across splits even when defined as "disjoint." Our identity-disjoint experiments demonstrate that convergence persists in aggregate even without overlap, but with more frequent divergences under identical seeds. This suggests that overlap is not strictly necessary but enhances alignment by providing near-redundant cues across groups.

The attribute splits further highlight the dataset's structure. When the partition is aligned with semantically strong attributes (eyeglasses, gender), the effective training distributions are less overlapping in identity and attribute space. Under these conditions, the same-noise alignment is substantially weaker. This indicates that the celebrated generalization is not a property of the UNet in the abstract, but a consequence of dataset homogeneity: CelebA's redundancy creates conditions under which inductive biases can align score functions across groups. In more heterogeneous domains, convergence is unlikely to be as pronounced.

6.3 Metric validity and methodological implications

A second methodological issue concerns how sample originality is evaluated. The original study relied on pixel-wise correlation to identify the nearest training example to a generated face. Our analysis shows that correlation is a misleading proxy: top-k correlation neighbors often include visually dissimilar faces, while perceptually closer candidates are overlooked. Re-ranking with LPIPS (VGG backbone) consistently changes the identity of the nearest neighbor, often revealing that a generated sample is almost a duplicate of a training image. In this light, claims that the models produce novel identities should be revisited: many generated samples are in fact closer to training instances than correlation-based metrics suggest.

The methodological implication is that evaluation of memorization vs. generalization in generative models requires perceptual metrics. Correlation, SSIM, or PSNR are insufficient for assessing semantic originality. LPIPS is not perfect—it inherits biases from its backbone—but it is substantially more aligned with human judgment. Our pipeline therefore incorporates LPIPS as a standard diagnostic and demonstrates that conclusions about generalization and sample novelty can shift significantly depending on metric choice.

6.4 Limitations

Several limitations constrain our conclusions. First, training was conducted at 40×40 resolution with 100,000 exposure repetitions rather than the 80×80 and 250,000 used in the original study, due to computational constraints. While qualitative phenomena were preserved, quantitative measures may differ at higher resolution and exposure. Second, LPIPS re-ranking was applied only to correlation-preselected top-k neighbors, not exhaustively over the full dataset. A streaming LPIPS computation across the entire CelebA tensor would provide stronger evidence but was computationally infeasible in our setting. Third, we restricted attribute-controlled splits to eyeglasses and gender, though CelebA offers 40 attributes; systematic exploration across multiple attributes could clarify how convergence scales with semantic divergence. Finally, these attribute-based experiments keep potential near-duplicates to preserve subset size and exact balance, which may slightly increase redundancy relative to the baseline and identity-disjoint settings.

7 Conclusion

This report has reproduced and extended the experiments of Kadkhodaie et al. Kadkhodaie et al. (2024) on memorization vs. generalization in diffusion models. We confirmed the central finding—that two UNet denoisers trained on disjoint subsets of CelebA can converge to nearly identical score functions—while also introducing methodological refinements to nuance this claim.

Our contributions are threefold. First, we developed a fully reproducible preprocessing pipeline with all the data available on GitHub, including deterministic ordering and memory-efficient duplicate removal, making large-scale training tractable without access to excessive hardware. Second, we challenged the evaluation protocol by introducing LPIPS-based nearest-neighbor analysis, showing that correlation systematically underestimates the similarity of generated samples to training data. Third, we designed attribute- and identity-controlled splits, demonstrating that convergence is weaker when datasets are semantically heterogeneous, and that identity overlap stabilizes but is not essential for alignment.

Taken together, these results suggest that the striking generalization reported in CelebA is partially an artifact of dataset structure and evaluation metrics. Inductive biases in UNet denoisers do promote alignment, but the extent of convergence depends strongly on redundancy in training data and on how originality is measured. Future work should extend this analysis to higher-resolution datasets, alternative architectures, and more perceptually grounded metrics to better characterize the balance between memorization and generalization in diffusion models.

References

S. Arora and Y. Zhang. Do gans actually learn the distribution? an empirical study, 2017. URL https://arxiv.org/abs/1706.08224.

A. Ghildyal and F. Liu. Shift-tolerant perceptual similarity metric, 2022. URL https://arxiv.org/abs/2207.13686.

J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models, 2020. URL https://arxiv.org/abs/2006.11239.

Z. Kadkhodaie, F. Guth, E. P. Simoncelli, and S. Mallat. Generalization in diffusion models arises from geometry-adaptive harmonic representation. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.02557.

Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015. URL https://arxiv.org/abs/1411.7766.

K. Miyasawa. An empirical bayes estimator of the mean of a normal population. *Bulletin de l'Institut International de Statistique*, 38, 01 1961.

M. Raphan and E. Simoncelli. Least squares estimation without priors or supervision. *Neural Computation*, 23:374–420, 02 2011. doi: 10.1162/NECO_a_00076.

H. E. Robbins. An empirical bayes approach to statistics. In *Proc Third Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 157–163, 1956.

G. Somepalli, V. Singla, M. Goldblum, J. Geiping, and T. Goldstein. Diffusion art or digital forgery? investigating data replication in diffusion models, 2022. URL https://arxiv.org/abs/2212.03860.

P. Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011. doi: 10.1162/NECO_a_00142.

R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. URL https://arxiv.org/abs/1801.03924.

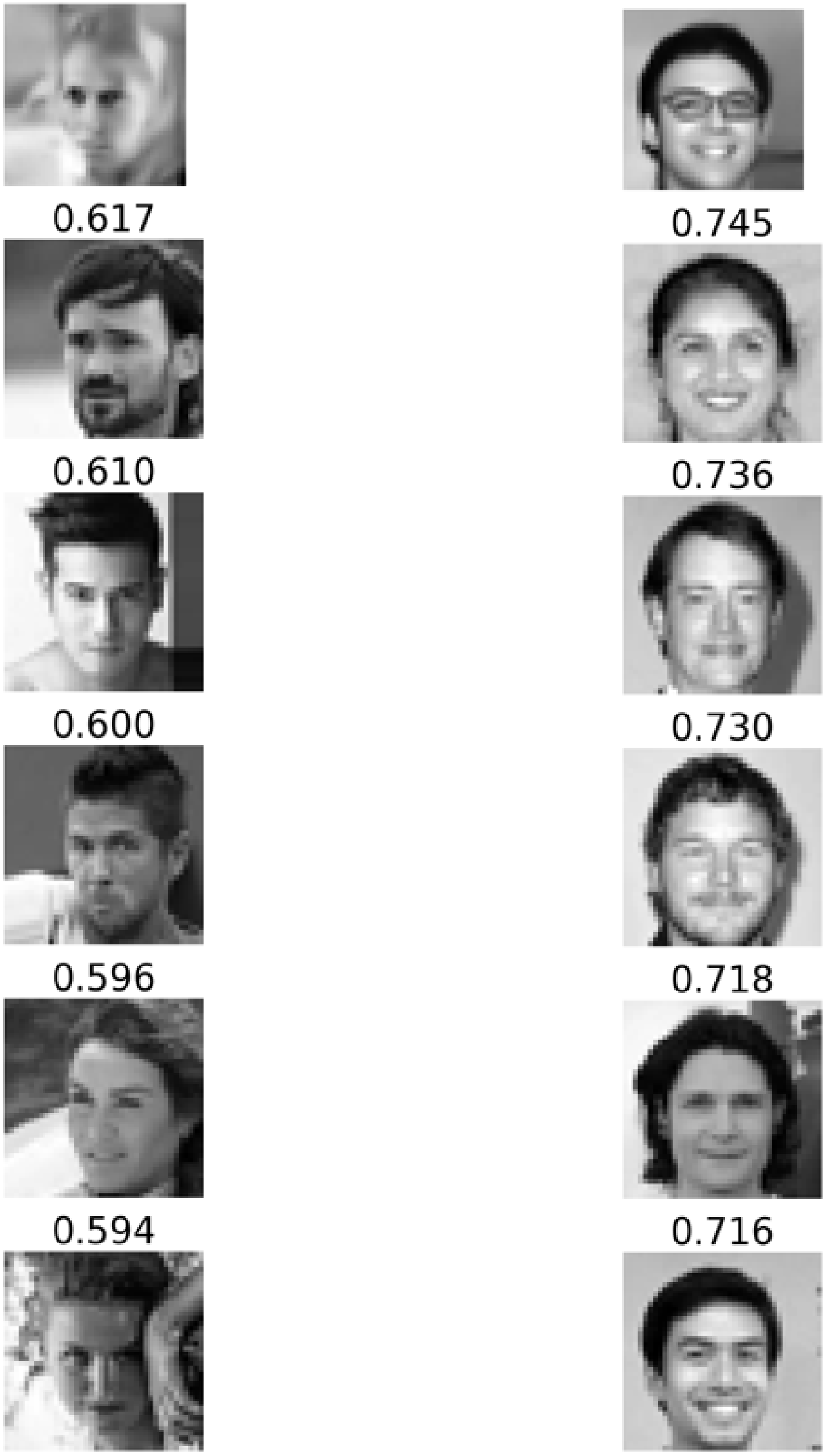
Acknowledgments

I would like to express my sincere gratitude to my supervisors, Vincent Duval and Antonin Chambolle, for their continuous guidance and support throughout this internship. Their expertise and insightful advice have been invaluable in the development of this project. I am also grateful for the warm welcome within the Mokaplan team at Inria, which made this experience both enriching and rewarding.

8 Appendix

For completeness, the full Python implementation of `remove_repeats_block` is provided below, and is also available together with the authors' original procedure in the `quality_metrics_func.py` file on GitHub:

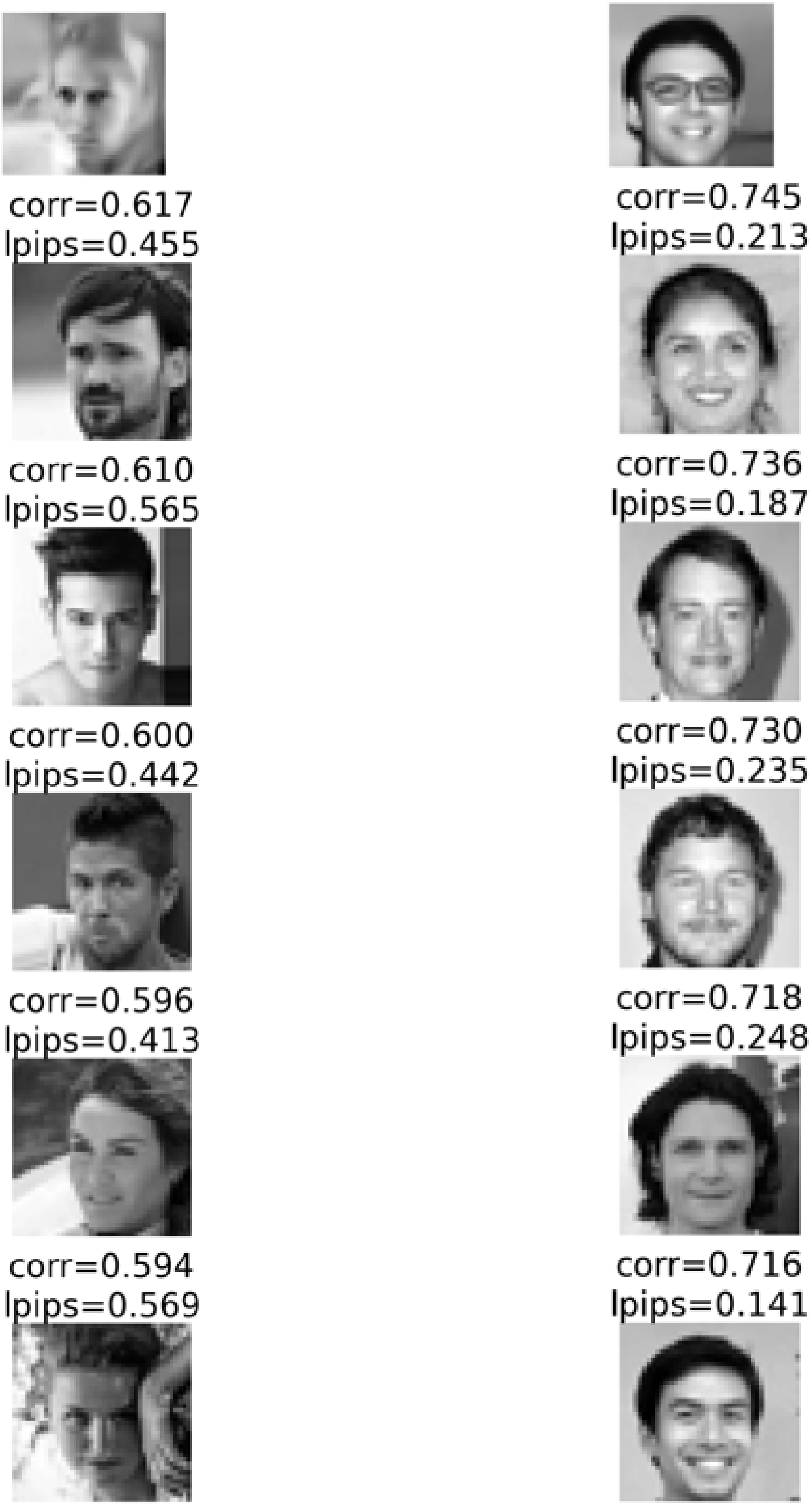
```python
def remove_repeats_block(dataset, threshold=0.95):

    remove_im_mean = lambda data : data - data.mean(dim=(1,2,3), keepdim=True)
    # Downsample to 20x20 for correlation computation
    pool = torch.nn.AvgPool2d(int(dataset.shape[2]/20))
    dataset_down = pool(dataset)
    dataset_down = remove_im_mean(dataset_down)

    # Normalize for correlation computation
    norms = dataset_down.norm(dim=(2,3), keepdim=True).norm(dim=1, keepdim=True)
    norms[norms == 0] = 0.001
    dataset_down = (dataset_down / norms).flatten(start_dim=1)

    N = dataset_down.shape[0]
    to_remove = []
    b = 5000  # Block size

    # Compute correlation matrix in blocks
    for i in range(0, N, b):
        bi = min(b, N-i)
        for j in range(0, i+1, b):
            bj = min(b, N-j)
            corrs = torch.matmul(dataset_down[i:i+bi],
                                 dataset_down[j:j+bj].T)
            rep_IDs_1, rep_IDs_2 = torch.where(torch.abs(corrs) > threshold)
            # Add duplicate indices to removal list
            for k in range(len(rep_IDs_1)):
                if rep_IDs_1[k] != rep_IDs_2[k]:  # Exclude diagonal
                    m = max(rep_IDs_1[k] + i, rep_IDs_2[k] + j)
                    if m not in to_remove:
                        to_remove.append(m)

    return np.delete(dataset, to_remove, axis=0)
```



(a) Example 1: correlation ranks a semantically inconsistent face first.

(b) Example 2: correlation selects an opposite-gender face as the top-1 neighbor.

Figure 10 – Correlation-based ranking of nearest neighbors at N=100,000. Each column shows the generated sample (top), followed by its top-k training neighbors according to pixel correlation. Correlation often prioritizes pixel-level alignment over semantic structure, yielding mismatches such as opposite-gender neighbors.



(a) Example 1: re-ranking corrects the cross-gender mismatch.

(b) Example 2: LPIPS retrieves a more perceptually coherent neighbor.

Figure 11 – LPIPS re-ranking of nearest neighbors at N=100,000 corresponding to Figure 10. Each column shows the generated sample (top), followed by its top-k training neighbors with LPIPS value (lower means more similar).