YOURS OR MINE? MISLEADING LATENT DIFFUSION MODEL ATTRIBUTION VIA CROSS-MODEL RECONSTRUCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reconstruction error-based attribution infers an image's source model from reconstruction errors under candidate autoencoders. We find that distinct Stable Diffusion models can produce identical errors for the same image, making source attribution ambiguous. Comparing autoencoder architectures and parameters reveals that this ambiguity arises from shared autoencoders. We therefore test whether reconstructing an image with another model's autoencoder can redirect attribution after generation. We introduce Cross-Model Reconstruction (CMR), which passes an image generated by the original model M_1 through the attack model M_2's encoder and decoder once. This reconstruction pass can induce false attribution to M_2 with minimal perceptual distortion and requires neither diffusion sampling nor attribution queries. Models that share an autoencoder can still have different denoisers. We propose READ (Responses Evidence for Attribution from Denoisers), which learns attribution from multiple denoisers' relative responses to controlled latent perturbations, using clean images for calibration. Across 12 generative models, CMR raises mean attack-model acceptance to 93.5-99.8% for AEDR, AEROBLADE, and LatentTracer. By comparing denoising responses, READ distinguishes source models even when they share an autoencoder. On the same attacked images, it reduces mean attack-model acceptance to 13.61% while improving acceptance of the original source model relative to these baselines. The source code of CMR and READ are included in the supplementary material.

1 INTRODUCTION

Model attribution aims to identify which latent diffusion model (LDM) originally generated an image, providing evidence for tracing generated content and protecting model copyright (Wang et al., 2023). Recent reconstruction error-based methods, such as AEDR (Wang et al., 2026a), AEROBLADE (Ricker et al., 2024), and LatentTracer (Wang et al., 2024), offer a practical approach because they require neither the original prompt nor recovery of the full diffusion trajectory. These methods build on the observation that a model's decoder can often accurately reproduce its own generated images. They reconstruct an input with candidate autoencoders and infer its source from the resulting errors.

We find that distinct Stable Diffusion models can produce identical reconstruction errors for the same image. Architecture and parameter comparisons trace this ambiguity to shared autoencoders, whose matching responses make their generators difficult to distinguish through reconstruction error (Ricker et al., 2024). Because an autoencoder can reconstruct images from other generators, we investigate whether passing an image from the original model M_1 through the attack model M_2's autoencoder can mislead attribution into accepting M_2 as its source.

We confirm this hypothesis using Cross-Model Reconstruction (CMR). CMR passes an image generated by M_1 through M_2's encoder and decoder once. The encoder supplies a latent code from which M_2's decoder reproduces the processed image, establishing reconstruction evidence favoring M_2 after generation. This single pass can mislead attribution into accepting M_2 as the source, even though M_1 originally generated the image. CMR preserves the image's main visual content and

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

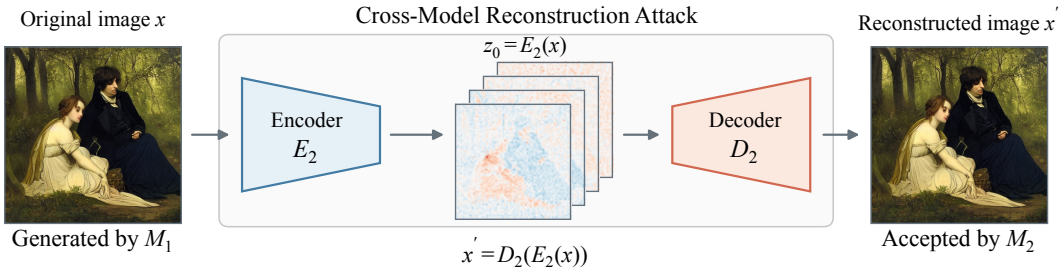


Figure 1: Overview of the Cross-Model Reconstruction (CMR) attack. An image x generated by the original model M_1 is encoded as $z_0 = E_2(x)$ and reconstructed as $x' = D_2(z_0)$ using the attack model M_2's autoencoder. By altering the reconstruction errors used for source attribution, CMR misleads acceptance of M_2 while maintaining perceptual similarity to x. The attack consists of a single encoding–decoding pass and requires neither diffusion sampling nor attribution queries.

requires neither diffusion sampling nor attribution queries. Paired JPEG and Gaussian-blur controls show that weakening original-source acceptance alone does not produce the same acceptance of M_2.

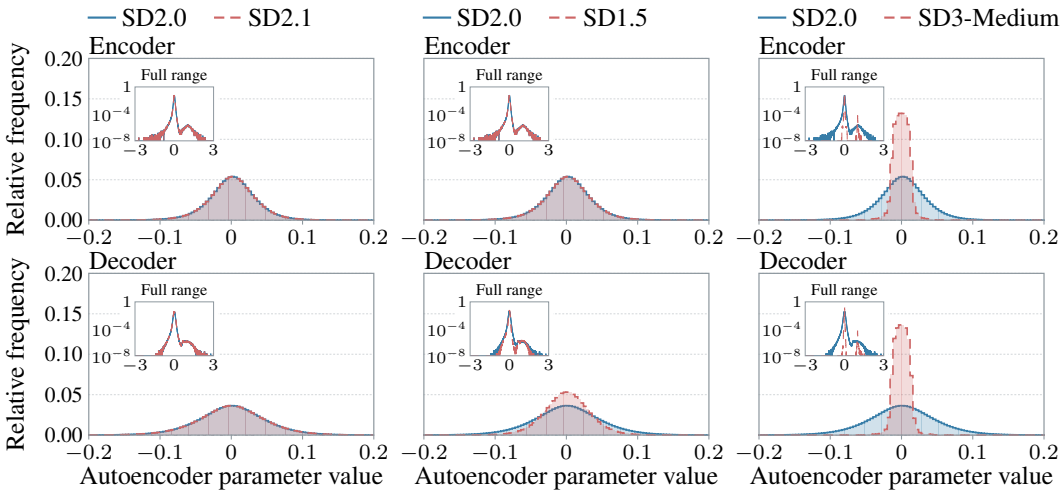
CMR changes the image through M_2's autoencoder, while its original generation also involved M_1's denoiser (Rombach et al., 2022). Models that share an autoencoder can have different denoisers, whose recovery responses may help distinguish their generated images. We therefore examine whether these responses retain information about the original generator after reconstruction. For the same input image, we compare how the candidate model and reference models recover their latent representations under controlled perturbations. The comparison measures the candidate's recovery behavior relative to other models across several noise levels.

We propose READ (Responses Evidence for Attribution from Denoisers), which learns source attribution from these relative responses. Controlled perturbations provide repeatable conditions for measuring recovery errors, and clean images with known sources calibrate the differences among models. READ normalizes the response vectors, compares the candidate with reference models, and learns an attribution rule using only these clean calibration images. We evaluate CMR and READ on 13,000 images, covering 12 generative models, 132 directed model pairs, and natural-image controls. CMR raises mean attack-model acceptance to 93.5–99.8% for AEDR, AEROBLADE, and LatentTracer. On the same attacked images, READ reduces this rate to 13.6% while improving recognition of the original source. Our contributions are threefold.

- We introduce CMR, a single-pass reconstruction attack that can redirect attribution toward the attack model while preserving the image's main visual content.
- We combine autoencoder architecture and parameter comparisons, reconstruction analysis, and paired processing controls to explain attribution ambiguity and how subsequent reconstruction creates evidence favoring the attack model.
- We propose READ, which learns source attribution from multiple denoisers' relative responses to controlled latent perturbations, improving attribution under reconstruction attacks and distinguishing models with shared autoencoders.

2 Background

Reconstruction error-based methods infer an image's source by measuring how accurately candidate autoencoders can reproduce it. AEROBLADE (Ricker et al., 2024) measures the perceptual error of a direct autoencoder reconstruction. LatentTracer (Wang et al., 2024) optimizes an encoder-initialized latent representation to reduce reconstruction error. AEDR (Wang et al., 2026a) uses the ratio of two successive reconstruction errors and calibrates it using image homogeneity. These methods require neither the original prompt nor recovery of the full diffusion trajectory. Autoencoder reconstruction has also been used to align real and generated images when training fake-image detectors (Rajan et al., 2025).



(a) Shared autoencoders (b) Same architecture, distinct weights (c) Different architectures

Figure 2: Distributions of all encoder (top row) and decoder (bottom row) parameters for three representative SD model pairs. (a) SD2.0 and SD2.1 share encoder and decoder architectures and weights. (b) SD1.5 and SD2.0 share the architecture and encoder weights, but their decoder weights differ. (c) SD2.0 and SD3-Medium use different architectures. Main panels show parameter values in $[-0.2, 0.2]$; insets show the full range with logarithmic frequency axes.

For a candidate model M_j with encoder E_j and decoder D_j, the reconstruction and its error are

$$\widehat{x}_j(x) = D_j(\widehat{z}_j(x)), \qquad e_j(x) = d(x, \widehat{x}_j(x)), \tag{1}$$

where $\widehat{z}_j(x)$ is obtained by encoding x and optionally refining the code, and d compares the input with its reconstruction. Each method converts this error, or errors from successive reconstructions, into a source score $s_j(x)$. The decision $b_j(x) = \mathbb{I}[s_j(x) \geq \tau_j]$ accepts M_j when the score reaches τ_j, a threshold calibrated on clean images and held fixed during attack evaluation.

Our architecture and parameter comparisons show that candidate models can share an autoencoder, use matching architectures with different weights, or use different architectures (Figure 2).

When two models share an autoencoder, the same deterministic reconstruction procedure gives $\widehat{x}_i(x) = \widehat{x}_j(x)$ and, for the same discrepancy measure, $e_i(x) = e_j(x)$. The errors then provide identical evidence for both candidates, consistent with Ricker et al. (2024). All three methods attain 50.00% mean clean AUROC for shared autoencoders, compared with 76.52–91.40% for different weights or architectures (Figure 5a). This reliance on the autoencoder leads us to test whether applying M_2's autoencoder after generation can mislead attribution of an image from M_1.

3 CMR: Cross-Model Reconstruction Attack

3.1 Threat Model

Given an image x generated by M_1, the attacker seeks a visually similar x' that is accepted as an output of M_2. The attacker has access to M_2's encoder E_2, decoder D_2, and required preprocessing. The attack needs neither M_1's parameters nor the original prompt, initial noise, or diffusion trajectory. Attribution rules and acceptance thresholds remain fixed during evaluation.

3.2 Cross-Model Reconstruction

CMR passes x through M_2's encoder and decoder once (Figure 1), obtaining the latent representation z_0 and reconstructed image x',

$$z_0 = E_2(x), \qquad x' = D_2(z_0). \tag{2}$$

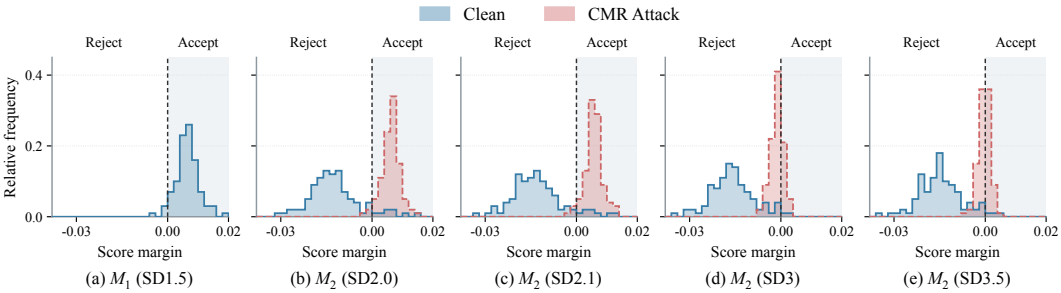


Figure 3: AEROBLADE score distributions before and after CMR. (a) Clean SD1.5 images evaluated under the original model M_1. (b–e) The same images, before and after reconstruction, evaluated under M_2: SD2.0, SD2.1, SD3-Medium, and SD3.5-Medium, respectively. Nonnegative margins indicate acceptance. Corresponding AEDR and LatentTracer distributions appear in Appendix D.3 (Figures 9 and 10).

This single pass requires no diffusion sampling, attribution queries, or iterative optimization. For SD1.5 images, CMR shifts AEROBLADE scores toward M_2's acceptance region (Figure 3). The model used for reconstruction thus receives stronger support as the source.

CMR introduces the code z_0, whereas attribution estimates a new code $\widehat{z}_2(x')$ from the reconstructed image. At the continuous decoder output, the error compares the two decoded images,

$$e_2(x') = d(x', \widehat{x}_2(x')) \quad = d(D_2(z_0), D_2(\widehat{z}_2(x'))) \,. \tag{3}$$

For a fixed feature map ϕ, consider the squared feature discrepancy

$$d(a, b) = \|\phi(a) - \phi(b)\|_2^2. \tag{4}$$

If $\phi \circ D_2$ is L_2-Lipschitz on a neighborhood containing both codes, then

$$e_2(x') \le L_2^2 \, \|\widehat{z}_2(x') - z_0\|_2^2 \,. \tag{5}$$

For fixed L_2, recovering a code close to z_0 gives a small error under M_2, even though M_1 remains the original source. Appendix B.2 proves this bound and extends it to quantized image outputs.

4 READ: Responses Evidence for Attribution from Denoisers

CMR changes reconstruction evidence through M_2's autoencoder, but the image was originally generated using M_1's denoiser. READ examines this component by comparing recovery errors from a candidate denoiser and $K - 1$ reference denoisers under controlled latent perturbations. Their relative responses provide features for a source classifier trained on clean images with known sources (Algorithm 1, Appendix C.2).

4.1 Denoising responses

Let $\mathcal{O}_j$ contain the candidate index j and $K - 1$ reference indices. For each $k \in \mathcal{O}_j$, we encode x as $z_0^{(k)} = E_k(x)$ and add noise at selected timesteps $t \in \mathcal{T}_k$,

$$z_t^{(k)} = \alpha_{k,t} z_0^{(k)} + \sigma_{k,t} \epsilon^{(k)}, \qquad \epsilon^{(k)} \sim \mathcal{N}(0, I), \tag{6}$$

where $\alpha_{k,t}$ and $\sigma_{k,t}$ follow the model's native noise schedule.

At each noise level, we add opposite perturbations $\pm\delta_{t,q}^{(k)}$ in direction q. Let $f_{k,t}$ predict the latent representation before noising. We compare the average of its two predictions with $z_0^{(k)}$, scaling the squared recovery error by the latent energy,

$$\begin{aligned} \bar{z}_{t,q}^{(k)} &= \frac{f_{k,t}(z_t^{(k)} + \delta_{t,q}^{(k)}) + f_{k,t}(z_t^{(k)} - \delta_{t,q}^{(k)})}{2}, \\ d_{t,q}^{(k)}(x) &= \frac{\mathrm{mean}[(\bar{z}_{t,q}^{(k)} - z_0^{(k)})^2]}{\max\{\mathrm{mean}[(z_0^{(k)})^2], \varepsilon_{\mathrm{num}}\}}, \end{aligned} \tag{7}$$

The means are over latent entries, and $\varepsilon_{\text{num}} > 0$ stabilizes normalization. For twice continuously differentiable $f_{k,t}$, symmetric averaging cancels the perturbation's first-order effect (Appendix B.3). Three noise levels and two directions give the response vector $\mathbf{d}^{(k)}(x) \in \mathbb{R}^6$.

4.2 Relative responses for attribution

We stack the responses into $T_j(x) \in \mathbb{R}^{K\times 6}$, with one row per model. A coordinate-wise normalizer g_j, fitted on clean calibration images, accounts for differences in latent scale and noise schedule. The resulting matrix $\widetilde{T}_j(x) = g_j(T_j(x))$ expresses each response relative to its calibration distribution.

Let $\widetilde{\mathbf{d}}^{(k)}(x)$ denote the normalized response vector of M_k. For each reference $k \in \mathcal{O}_j \setminus \{j\}$, we compute $\widetilde{\mathbf{d}}^{(k)}(x) - \widetilde{\mathbf{d}}^{(j)}(x)$. A positive entry indicates that the candidate has a lower normalized recovery error for that measurement. The feature map Ψ_j combines these differences with the normalized responses, summary statistics, and the candidate's ranks among models. We train a gradient-boosted classifier h_j on clean images, using M_j outputs as positives and other sources as negatives. It maps the resulting features to a source score,

$$\mathbf{v}_j(x) = \Psi_j\left(\widetilde{T}_j(x)\right), \qquad s_j(x) = h_j(\mathbf{v}_j(x)). \tag{8}$$

READ accepts M_j when $s_j(x) \geq \tau_j$. Reference models, normalizers, classifiers, and thresholds are selected or fitted using only clean calibration images and held fixed during attack evaluation (Appendix C).

5 Experiments

5.1 Experimental Setup

Generative Models. We evaluate 12 generative models: SD1.5, SD2.0, SD2.0-base, SD2.1, SD2.1-base, SDXL-base, SDXL-Turbo, SD3-Medium, SD3.5-Medium, SD3.5-Large, Kandinsky 2.1 (KD2.1), and VQdiffusion (VQDM). The SD3 and SD3.5 models use rectified flows and multimodal diffusion transformers (Esser et al., 2024; Stability AI, 2024), while VQDM uses discrete latent representations. Appendix A.2 groups these models by autoencoder architecture and parameter sharing.

Attribution Methods. We compare READ with the reconstruction baselines AEDR (Wang et al., 2026a), AEROBLADE (Ricker et al., 2024), and LatentTracer (Wang et al., 2024). We also evaluate fingerprint matching with Model-Fingerprint (Yang et al., 2025), retrieval with LIDA (Wang et al., 2026b), and learned forensic representations with Forensic Self-Descriptions (FSD) (Nguyen et al., 2025). Attribution rules, reference galleries, and thresholds remain fixed during attack testing (Appendix C).

Dataset. Our dataset contains 13,000 images: 1,000 natural images from LAION-Aesthetics (Schuhmann et al., 2022) and 1,000 images from each of the 12 generators. To reduce content differences across sources, we extract prompts from the natural images using CLIP Interrogator[1] and use them for every generator. Each source contributes 500 calibration and 500 test images. Each attribution test pairs 500 candidate-model images with 500 images from another generator or the natural-image set (Appendix A).

Metrics. For generated images, we report accuracy and AUROC for M_1 attribution, alongside M_1 and M_2 acceptance rates. For natural images, we report accuracy, AUROC, false-positive rate (FPR), and true-positive rate (TPR). We assess CMR reconstruction fidelity using PSNR, SSIM (Wang et al., 2004), LPIPS (Zhang et al., 2018), and DINOv2 cosine similarity (Oquab et al., 2024). We average generated-image results equally over model pairs within each autoencoder group and natural-image results equally over all 12 candidates. We run all experiments with PyTorch on NVIDIA GeForce RTX 4090 GPUs and report means over three random seeds. Metric and fidelity details appear in Appendices A and D.

[1] https://github.com/pharmapsychotic/clip-interrogator

Table 1: Generated-image attribution after CMR. Shared AE, Same arch., and Diff. arch. denote shared autoencoders, matching architectures with different parameters, and different architectures. Attribution metrics (%) are averaged equally over model pairs within each group; parentheses give percentage-point changes from Clean. Image-quality scores compare each reconstruction with its input, follow the same averaging, and are shared across methods (Appendix D.2).

Method	Accuracy	AUROC	M_1 accept	M_2 accept ↓	Image quality	
Shared AE						
AEDR	50.60 (+0.41)	84.17 (+34.17)	100.00 (+0.83)	100.00 (+1.20)	PSNR	34.26
AEROBLADE	51.26 (+1.92)	69.32 (+19.32)	97.67 (+3.84)	98.44 (+3.28)	LPIPS	0.0204
LatentTracer	56.61 (+5.93)	78.22 (+28.22)	99.14 (+11.85)	98.79 (+12.86)	DINO	0.9950
Same arch.						
AEDR	51.41 (−26.77)	63.13 (−13.39)	45.08 (−53.54)	99.86 (+57.60)	PSNR	29.44
AEROBLADE	59.06 (−23.86)	65.27 (−20.12)	44.82 (−47.71)	96.31 (+69.62)	LPIPS	0.0390
LatentTracer	57.00 (−20.68)	68.66 (−11.53)	45.76 (−41.36)	98.65 (+66.89)	DINO	0.9862
Diff. arch.						
AEDR	52.39 (−28.86)	63.44 (−21.89)	38.72 (−57.73)	99.86 (+65.92)	PSNR	29.53
AEROBLADE	54.97 (−22.54)	62.93 (−24.36)	42.06 (−45.09)	91.69 (+59.56)	LPIPS	0.0388
LatentTracer	53.20 (−23.24)	60.18 (−31.22)	34.93 (−46.47)	98.02 (+69.50)	DINO	0.9847

Table 2: Natural-image detection after CMR. Metrics (%) are averaged equally over candidates; parentheses give percentage-point changes from Clean. CMR reconstructs natural negatives while generated positives and thresholds remain fixed. FPR measures erroneous acceptance of natural images. Image-quality scores are shared across methods (Appendix D.2); results by autoencoder type appear in Table 8.

Method	Accuracy	AUROC	FPR	Image quality	
AEDR	49.19 (−36.31)	32.33 (−53.54)	98.88 (+72.62)	PSNR	29.42
AEROBLADE	55.48 (−33.41)	54.68 (−37.70)	85.63 (+66.82)	LPIPS	0.0621
LatentTracer	48.83 (−34.17)	29.85 (−55.40)	95.25 (+68.33)	DINO	0.9748

5.2 Misleading of the CMR Attack on Attribution Performance

Generated images. For model pairs with different autoencoder parameters or architectures, CMR raises M_2 acceptance to 91.69–99.86% and reduces M_1 acceptance to 34.93–45.76% (Table 1). Accuracy drops by 20.68–28.86 percentage points and AUROC by 11.53–31.22 points, showing that CMR disrupts score ranking as well as decisions at fixed thresholds. CMR introduces a code from which M_2's decoder can reproduce the attacked image, providing reconstruction evidence acquired after generation. Shared autoencoders instead leave the source unresolved because both candidates are accepted at nearly 100%. Their higher AUROC separates reconstructed M_1 positives from fixed clean M_2 negatives. The common reconstruction map separates these processing conditions while continuing to support both candidate sources.

Natural images. CMR raises mean FPR to 85.63–98.88% across the three reconstruction baselines and reduces mean Accuracy by 33.41–36.31 percentage points (Table 2). Generated positives and thresholds remain unchanged, so this loss comes from falsely accepting reconstructed natural images. For AEDR and LatentTracer, AUROC falls below 50%, with reconstructed natural negatives tending to rank above generated positives. Reconstruction alone can therefore make natural images score as stronger source matches than the model's own outputs.

Image fidelity. Mean LPIPS is 0.0369 for generated images and 0.0621 for natural images, with DINO similarities of 0.9862 and 0.9748, respectively (Tables 1 and 2). Fidelity is highest with shared autoencoders; the other groups retain 29.44–29.53 dB mean PSNR alongside high M_2 acceptance. High attack-model acceptance therefore coexists with preserved visual content, as illustrated by the retained subjects and scene layout in Figure 4.

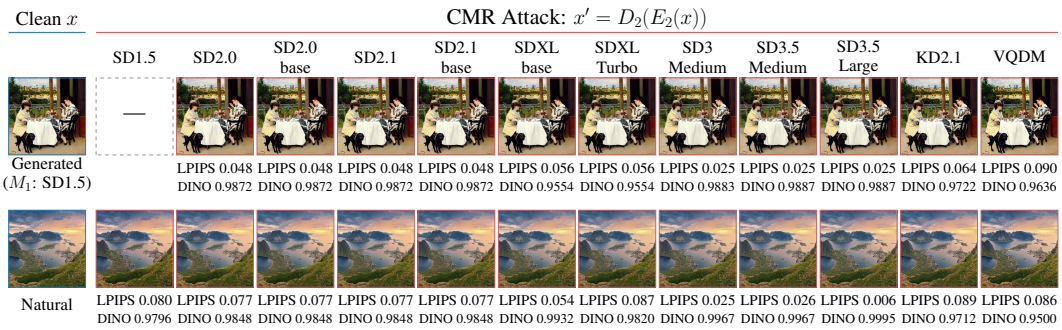
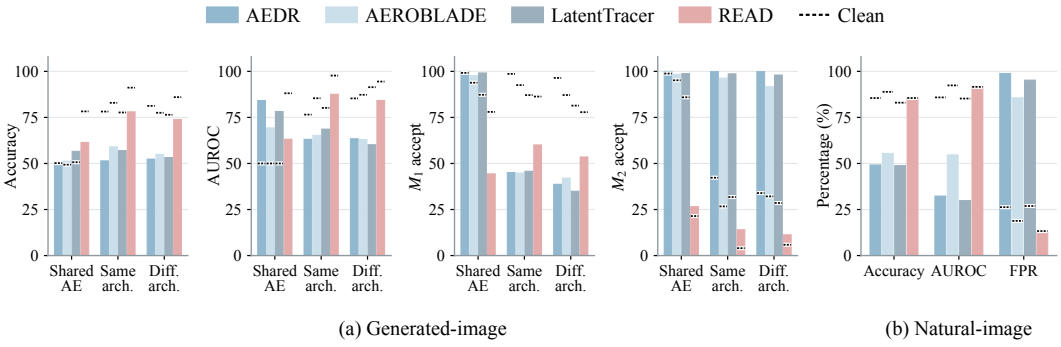


Figure 4: CMR reconstructions of a generated image and a natural image. LPIPS and DINO compare each output with its corresponding input. Details in Appendix D.2.



(a) Generated-image (b) Natural-image

Figure 5: Attribution performance of READ and three reconstruction baselines.

5.3 Performance of the Proposed READ Method

Comparing with three autoencoder baselines. READ improves Accuracy across all autoencoder groups (Figure 5a). For different autoencoders, it improves AUROC and M_1 acceptance while reducing M_2 acceptance from 91.69–99.86% to 11.28–14.04%. In these groups, READ retains more of its clean AUROC after CMR. For shared autoencoders, the baselines accept both candidates at nearly 100%, whereas READ accepts M_1 and M_2 at 44.36% and 26.54%, respectively. The difference in acceptance shows that READ's source decisions remain selective even when the candidates share their reconstruction map. On reconstructed natural images (Figure 5b), READ limits false acceptance to 12.62%, compared with 85.63–98.88% for the baselines. Accuracy and AUROC remain close to Clean at 85.87% and 91.84%. Denoising responses retain the distinction between candidate-generated images and natural images subsequently processed by its autoencoder.

Comparing with additional baselines. Table 3 reports Model-Fingerprint, LIDA, and FSD on their associated public data (Appendix C.5). READ uses the 12-model benchmark; parentheses give percentage-point differences from its mean across the different evaluation settings. Under CMR, Model-Fingerprint's AUROC falls from 92.04% to 54.21%, and M_2 acceptance reaches 79.17%, compared with 20.83% for M_1. CMR can thus redirect fingerprint-based attribution as well. LIDA and FSD accept M_2 less often, alongside M_1 acceptance of 23.50% and 9.50%, respectively. LIDA's clean AUROC of 48.23% indicates source confusion before CMR, whereas FSD's M_1 acceptance falls from 53.00% to 9.50% after reconstruction. Their post-CMR results thus reflect pre-existing source confusion for LIDA and a loss of original-source recognition for FSD.

5.4 In-depth Analysis of CMR and READ

Reconstruction evidence under CMR. CMR yields higher M_2 acceptance than JPEG or Gaussian blur across the same 12 attack models (Table 4). For AEDR, JPEG and CMR give similar M_1 acceptance at 45.44% and 46.86%, yet M_2 acceptance differs sharply at 30.46% and 99.87%, respectively. For AEROBLADE, JPEG weakens M_1 acceptance more than CMR without compa-

Table 3: Comparison with additional attribution baselines.

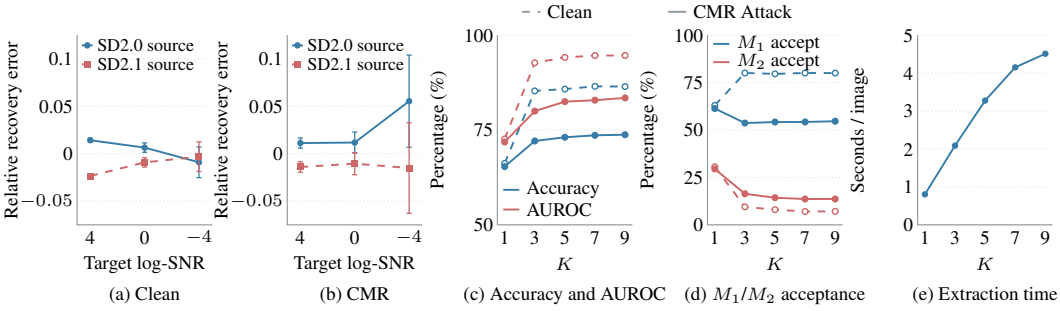
Method	Accuracy	AUROC	M_1 accept	M_2 accept
READ	73.60	82.86	54.24	13.61
Model-Fingerprint	54.17 (−19.43)	54.21 (−28.65)	20.83 (−33.41)	79.17 (+65.56)
LIDA	49.75 (−23.85)	51.66 (−31.20)	23.50 (−30.74)	24.00 (+10.39)
FSD	48.25 (−25.35)	44.63 (−38.23)	9.50 (−44.74)	23.50 (+9.89)

rable M_2 acceptance; Gaussian blur keeps M_2 acceptance close to Clean for both baselines. Both perturbation controls have lower mean LPIPS than CMR (Appendix C.6). These controls support a targeted effect in which CMR strengthens evidence for M_2 as well as weakening evidence for M_1.

Table 4: Attribution under CMR, JPEG compression, and Gaussian blur (%). Results are averaged equally over model pairs using the same test images and fixed thresholds. Perturbation selection and image quality are detailed in Appendix C.6.

Method	Condition	Accuracy	AUROC	M_1 accept	M_2 accept
AEDR	Clean	77.17	79.31	97.30	42.96
	CMR Attack	51.95	65.56	46.86	99.87
	JPEG	51.24	66.33	45.44	30.46
	Gaussian blur	72.32	75.68	87.60	41.13
AEROBLADE	Clean	75.92	82.85	89.25	37.41
	CMR Attack	55.63	64.21	48.67	93.59
	JPEG	47.11	51.28	31.63	23.13
	Gaussian blur	75.16	82.72	87.74	41.65

Source discrimination with shared autoencoders. We compare denoising responses for SD2.0 and SD2.1, whose shared autoencoder gives identical reconstruction errors. At low noise, images from each source have lower mean normalized recovery error under that source's denoiser (Figure 6a). After both image sets pass through the shared autoencoder, the gap narrows while retaining its source-dependent direction (Figure 6b). For this model pair, the relative denoising errors therefore retain source information after the common reconstruction step.



(a) Clean (b) CMR (c) Accuracy and AUROC (d) M_1/M_2 acceptance (e) Extraction time

Figure 6: Denoising responses and reference-model ablation. (a,b) Relative recovery errors for SD2.0 and SD2.1 images before and after CMR (means ± 1.96 standard errors). (c) Accuracy and AUROC, (d) source acceptance, and (e) estimated response-extraction time per image, averaged equally over model pairs. K counts the candidate and reference models. Dashed and solid lines in (c,d) denote Clean and CMR Attack, respectively. Measurement details appear in Appendix C.6.

5.5 Ablation Studies and Efficiency

Reference models and efficiency. We evaluate nested model sets with $K \in \{1, 3, 5, 7, 9\}$ (Figure 6c–e). Adding the first two references raises clean AUROC from 72.63% to 92.72%, supporting attribution based on the candidate's responses relative to other models. Increasing K from 7 to 9

adds only 0.03 and 0.60 percentage points to clean and post-CMR AUROC, respectively. Estimated response-extraction time rises from 0.81 seconds per image at $K = 1$ to 4.15 at $K = 7$ and 4.51 at $K = 9$. We use $K = 7$ to retain most of the attribution gain at a lower extraction cost. Timing averages the cost of all models needed for both source decisions; measurement scope and initialization overhead appear in Appendix C.6.

Noise and attribution features. At $K = 7$, combining three noise levels raises post-CMR AUROC from 75.19–76.18% to 82.86% and lowers M_2 acceptance under the same calibration and classifier settings (Table 5). Each noise level tests recovery at a different corruption strength, allowing the classifier to compare models across several recovery conditions. Relative features make the candidate's differences and ranks among models explicit to the classifier. They improve clean AUROC from 82.86% to 94.62%, with a smaller post-CMR gain from 81.05% to 82.86%. Under CMR, the classifier rejects M_2 more often, with M_1 acceptance decreasing from 58.91% to 54.24% at the fixed thresholds.

Table 5: READ component ablations under CMR (%, $K = 7$). Results are averaged equally over model pairs; parentheses give changes from each variant's Clean performance in percentage points. Ablation settings appear in Appendix C.6.

Configuration	Accuracy	AUROC	M_1 accept	M_2 accept ↓
Low noise only	70.26 (−11.63)	76.18 (−13.26)	52.50 (−23.27)	19.63 (+7.65)
Intermediate noise only	69.14 (−7.30)	75.21 (−8.50)	57.97 (−14.61)	21.25 (+1.56)
High noise only	67.31 (−1.89)	75.19 (−1.02)	61.67 (−3.78)	27.21 (+0.16)
Response vectors only	71.51 (−2.04)	81.05 (−1.82)	58.91 (−4.08)	15.62 (−0.27)
READ	73.60 (−12.90)	82.86 (−11.76)	54.24 (−25.80)	13.61 (+6.57)

Adding reconstruction scores. Combining READ with AEDR or AEROBLADE through score or feature fusion, calibrated on clean images, raises clean AUROC to as high as 96.87% from READ's 94.62% (Table 6). Under CMR, score fusion raises AUROC to 83.50–83.57% and M_2 acceptance to 20.08–23.62% from 13.61%. READ retains the highest Accuracy and M_1 acceptance and the lowest M_2 acceptance among these configurations. The higher false acceptance is consistent with fusion using reconstruction cues that favor M_2 after CMR.

Table 6: Fusion of READ with reconstruction scores under CMR (%, $K = 7$). Results are averaged equally over model pairs; parentheses give changes from each configuration's Clean performance in percentage points. Fusion details appear in Appendix C.4.

Configuration	Accuracy	AUROC	M_1 accept	M_2 accept ↓
READ	73.60 (−12.90)	82.86 (−11.76)	54.24 (−25.80)	13.61 (+6.57)
+ AEDR (score fusion)	71.44 (−18.19)	83.57 (−12.63)	48.62 (−36.37)	23.62 (+17.88)
+ AEDR (feature fusion)	63.81 (−26.04)	77.02 (−19.10)	32.40 (−52.07)	34.55 (+29.76)
+ AEROBLADE (score fusion)	73.07 (−15.28)	83.50 (−12.43)	52.15 (−30.57)	20.08 (+14.07)
+ AEROBLADE (feature fusion)	67.01 (−22.76)	79.20 (−17.67)	38.42 (−45.53)	29.46 (+25.07)

6 Discussion and conclusion

CMR shows that a single pass through another model's autoencoder can redirect source attribution while preserving visual content. This reconstruction supplies evidence favoring the attack model after generation, while shared autoencoders give identical responses for distinct generators.

READ uses relative denoising responses to recover source information after CMR, including for generators that share an autoencoder. Calibration uses labeled clean images and a fixed model set, with attribution rules frozen during attack evaluation. READ reduces attack-model acceptance and improves original-source recognition relative to reconstruction-based baselines. READ also yields lower strict reattribution attack success rates than the tested fusion variants under attack-model selection and further processing (Appendix D.4). These findings support denoising responses as source evidence that remains informative after reconstruction.

AI USE STATEMENT

We used AI tools to assist with manuscript drafting and language polishing, as well as literature retrieval and discovery. We also used AI to organize, test, and document the supplementary code. We take full responsibility for the manuscript and associated code, including all claims, results, and references.

REPRODUCIBILITY STATEMENT

Section 5.1 and Appendix A describe the datasets, generative models, calibration and test splits, and evaluation metrics. Appendix B provides the assumptions and proofs for the theoretical analysis, and Appendix C specifies the implementation details and hyperparameters of CMR, READ, and the comparison methods. The supplementary material includes the core implementations of CMR and READ, together with an README documenting installation, response extraction, classifier training, and inference.

REFERENCES

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12606–12633. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/esser24a.html.

Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 10686–10696. IEEE, 2022. doi: 10.1109/CVPR52688.2022.01043. URL https://doi.org/10.1109/CVPR52688.2022.01043.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html.

Tai D. Nguyen, Aref Azizpour, and Matthew C. Stamm. Forensic self-descriptions are all you need for zero-shot detection, open-set source attribution, and clustering of AI-generated images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3040–3050, 2025. URL https://arxiv.org/abs/2503.21003.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 2024, 2024. URL https://openreview.net/forum?id=a68SUt6zFt.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, 12:2825–2830, 2011. doi: 10.5555/1953048.2078195. URL https://dl.acm.org/doi/10.5555/1953048.2078195.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=di52zR8xgf.

Anirudh Sundara Rajan, Utkarsh Ojha, Jedidiah Schloesser, and Yong Jae Lee. Aligned datasets improve detection of latent diffusion-generated images. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=doBkiqESYq.

Anton Razzhigaev, Arseniy Shakhmatov, Anastasia Maltseva, Vladimir Arkhipkin, Igor Pavlov, Ilya Ryabov, Angelina Kuts, Alexander Panchenko, Andrey Kuznetsov, and Denis Dimitrov. Kandinsky: An improved text-to-image synthesis with image prior and latent diffusion. In Yansong Feng and Els Lefever (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023 - System Demonstrations, Singapore, December 6-10, 2023*, pp. 286–295. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-DEMO.25. URL https://doi.org/10.18653/v1/2023.emnlp-demo.25.

Jonas Ricker, Denis Lukovnikov, and Asja Fischer. AEROBLADE: Training-free detection of latent diffusion images using autoencoder reconstruction error. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 9130–9140. IEEE, 2024. doi: 10.1109/CVPR52733.2024.00872. URL https://doi.org/10.1109/CVPR52733.2024.00872.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 10674–10685. IEEE, 2022. doi: 10.1109/CVPR52688.2022.01042. URL https://doi.org/10.1109/CVPR52688.2022.01042.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=TIdIXIpzhoI.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In Ales Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part LXXXVI*, volume 15144 of *Lecture Notes in Computer Science*, pp. 87–103. Springer, 2024. doi: 10.1007/978-3-031-73016-0_6. URL https://doi.org/10.1007/978-3-031-73016-0_6.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: An open large-scale dataset for training next generation image-text models. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/a1859debfb3b59d094f3504d5ebb6c25-Abstract-Datasets_and_Benchmarks.html.

Stability AI. Introducing Stable Diffusion 3.5, October 2024. URL https://stability.ai/news-updates/introducing-stable-diffusion-3-5.

Chao Wang, Zijin Yang, Yaofei Wang, Weiming Zhang, and Kejiang Chen. AEDR: Training-free AI-generated image attribution via autoencoder double-reconstruction. In Sven Koenig, Chad Jenkins, and Matthew E. Taylor (eds.), *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pp.

9675–9683. AAAI Press, 2026a. doi: 10.1609/AAAI.V40I12.37930. URL https://doi.org/10.1609/aaai.v40i12.37930.

Hongsong Wang, Renxi Cheng, Chaolei Han, and Jie Gui. Attribution as retrieval: Model-agnostic AI-generated image attribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14062–14072, 2026b. URL https://openaccess.thecvf.com/content/CVPR2026/html/Wang_Attribution_as_Retrieval_Model-Agnostic_AI-Generated_Image_Attribution_CVPR_2026_paper.html.

Zhenting Wang, Chen Chen, Yi Zeng, Lingjuan Lyu, and Shiqing Ma. Where did I come from? origin attribution of AI-generated images. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/ebb4c188fafe7da089b41a9f615ad84d-Abstract-Conference.html.

Zhenting Wang, Vikash Sehwag, Chen Chen, Lingjuan Lyu, Dimitris N. Metaxas, and Shiqing Ma. How to trace latent generative model generated images without artificial watermark? In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 51396–51414. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/wang24bj.html.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4):600–612, 2004. doi: 10.1109/TIP.2003.819861. URL https://doi.org/10.1109/TIP.2003.819861.

Tianyun Yang, Juan Cao, Danding Wang, and Chang Xu. Model synthesis for zero-shot model attribution. *IEEE Transactions on Multimedia*, 2025. URL https://arxiv.org/abs/2307.15977.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 586–595. Computer Vision Foundation / IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00068. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Zhang_The_Unreasonable_Effectiveness_CVPR_2018_paper.html.

A EXPERIMENTAL SETUP AND EVALUATION METRICS

A.1 DATASET AND CALIBRATION

Natural images are drawn from the LAION-Aesthetics collection (Schuhmann et al., 2022).[2] Prompts are extracted from the LAION-derived images using CLIP Interrogator in classic mode with the ViT-H-14/laion2b_s32b_b79k CLIP model.

A.2 AUTOENCODER ARCHITECTURES AND PARAMETER SHARING

Table 7 summarizes the autoencoders used by the 12 evaluated generators. The evaluated SD3.5 checkpoints follow the released model configurations.[3] We compare module types, block widths, latent channels, projection layers, and parameter tensor shapes to identify matching architectures. Parameter sharing is established by comparing corresponding tensor names, shapes, and values in the evaluated checkpoints. Encoder and decoder parameters are also compared separately to identify partial sharing.

Table 7: Autoencoder configurations of the evaluated generators. Block widths follow the encoder from image to latent; decoder stages use the reverse order. The VQDM channel count describes the encoder output before codebook projection.

Model	Autoencoder	Channels	Block widths
SD1.5	KL	4	128, 256, 512, 512
SD2.0	KL	4	128, 256, 512, 512
SD2.0-base	KL	4	128, 256, 512, 512
SD2.1	KL	4	128, 256, 512, 512
SD2.1-base	KL	4	128, 256, 512, 512
SDXL-base	KL	4	128, 256, 512, 512
SDXL-Turbo	KL	4	128, 256, 512, 512
SD3-Medium	KL	16	128, 256, 512, 512
SD3.5-Medium	KL	16	128, 256, 512, 512
SD3.5-Large	KL	16	128, 256, 512, 512
KD2.1	MoVQ	4	128, 256, 256, 512
VQDM	VQ	256	64, 128, 256, 512

Architectural differences. The SD1.x and SD2.x autoencoders (Rombach et al., 2022), SDXL-base (Podell et al., 2024), and SDXL-Turbo (Sauer et al., 2024) share a four-stage architecture with four latent channels and learned projections around the latent bottleneck. The SD3 family increases the latent channel count to 16 and removes these projection layers. KD2.1 (Razzhigaev et al., 2023) uses a MoVQ autoencoder with four latent channels and a 16,384-entry codebook. VQDM (Gu et al., 2022) projects its 256-channel encoder output to 128-dimensional embeddings and quantizes them using a 4,096-entry codebook. Both quantized autoencoders include attention in the final encoder stage and the first decoder stage, with block widths that differ from the KL models.

Shared autoencoders. SD2.0, SD2.0-base, SD2.1, and SD2.1-base share all encoder and decoder parameters. SD3.5-Medium and SD3.5-Large form a second group with shared autoencoder parameters.

Same architecture, different parameters. The seven 4-channel KL models share an architecture, while their complete autoencoder weights form four distinct sets: SD1.5, the SD2 group, SDXL-base, and SDXL-Turbo. SD1.5 and the SD2 models share encoder parameters, but their decoder parameters differ. SDXL-base and SDXL-Turbo differ in both encoder and decoder parameters in the evaluated checkpoints. Within the 16-channel KL family, SD3-Medium has the same architecture as the two SD3.5 models, but different encoder and decoder parameters. These comparisons

[2] LAION-Aesthetics image collection, revision `3abfff0`.

[3] Official model cards: SD3.5-Medium and SD3.5-Large.

separate complete autoencoder sharing from matching architectures or shared encoders alone: different decoder parameters can change the reconstruction even when the encoded representation is identical.

Different architectures. These families are the 4-channel KL, 16-channel KL, KD2.1 MoVQ, and VQDM autoencoders. The distinction between 4-channel KL and KD2.1, for example, follows their module types, block widths, and quantization components despite their equal latent channel counts.

A.3 Evaluation protocol and metric aggregation

In the 12-model benchmark, each generated-image evaluation uses 500 positive images from the original model M_1 and 500 negative images from the paired model M_2. CMR reconstructs the positives with M_2's autoencoder, while the negatives remain unchanged. Natural-image evaluation pairs 500 generated positives with 500 LAION-derived natural negatives for each candidate. CMR reconstructs the natural negatives with the attack model's autoencoder, while the generated positives remain unchanged.

For a metric m_r computed on ordered model pair r, the reported mean over a group of R pairs is $\bar{m} = R^{-1} \sum_{r=1}^{R} m_r$. For each pair, Accuracy and AUROC evaluate M_1 using its images as positives and the paired model's images as negatives. Accuracy is $(\mathrm{TP} + \mathrm{TN})/(\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN})$. AUROC is the probability that a randomly chosen positive receives a higher source score than a randomly chosen negative, with half credit for tied scores. Section 5 reports equal-weight means of these per-pair metrics, including AUROC. Natural-image FPR is calculated for each attack model on its own reconstructed reference images and averaged equally over models.

The M_1 and M_2 acceptance rates evaluate both candidates on the same M_1 images, using x for Clean and x' for CMR. For natural images, FPR measures acceptance of natural negatives and TPR measures acceptance of clean images generated by the candidate. With balanced positive and negative classes, Accuracy equals $(\mathrm{TPR}+1-\mathrm{FPR})/2$. For Clean-to-CMR comparisons, the signed change is $\Delta = 100(\bar{m}_{\mathrm{CMR}} - \bar{m}_{\mathrm{Clean}})$ percentage points, where $\bar{m}$ is the corresponding mean metric expressed as a fraction. Differences are calculated before rounding the displayed values.

A.4 Joint acceptance and strict reattribution

The pair (b_1, b_2) gives four outcomes: M1-only $(1, 0)$, M2-only $(0, 1)$, Both $(1, 1)$, and Neither $(0, 0)$. These outcomes record acceptance of the original model and attack model separately. For N evaluated positive image–model pairs, the strict reattribution attack success rate is

$$\mathrm{RTASR}_{\mathrm{strict}} = \frac{1}{N} \sum_{i=1}^{N} \underbrace{b_1(x_i)[1 - b_2(x_i)]}_{\text{M1-only before attack}} \underbrace{[1 - b_1(x_i')]b_2(x_i')}_{\text{M2-only after attack}}. \tag{9}$$

The denominator includes all evaluated positive image–model pairs; fixed-failure recovery is defined in Appendix D.5. Attack-model acceptance is $N^{-1} \sum_i b_2(x_i')$.

B Theoretical analysis

B.1 Shared autoencoders and source discrimination

Let $A_j = D_j \circ E_j$ denote the deterministic reconstruction map of candidate model M_j.

Proposition 1 (Candidate responses with identical components) *If two candidate autoencoders satisfy $A_i = A_j$ as deterministic maps and use the same feature map h, then $r_i(x) = h(x, A_i(x)) = h(x, A_j(x)) = r_j(x)$. If their decision maps and thresholds are also identical, their acceptance decisions agree for every x, precluding exclusive acceptance of either candidate.*

Proof of Proposition 1. Since $A_i = A_j$, we have $A_i(x) = A_j(x)$ for every input x. Applying the same deterministic map h gives $r_i(x) = r_j(x)$. Identical decision maps and thresholds then produce identical acceptance decisions, with both candidates accepted or both rejected.

For iterative reconstruction features, identical initial inputs and the same deterministic autoencoder at every step yield identical iterates. For latent optimization, identical responses additionally require the same initialization, objective, optimizer, and randomness. Candidate-specific thresholds or supervised decision functions can produce different acceptance decisions.

Lemma 1 (AUROC under exchanged source labels) *Let s be a fixed real-valued scoring function, with independent $X \sim P$ and $Y \sim Q$, and give ties half credit in AUROC. Exchanging the positive and negative classes without changing score orientation gives*

$$\mathrm{AUC}(s; P, Q) + \mathrm{AUC}(s; Q, P) = 1. \tag{10}$$

Proof of Lemma 1. Write $U = s(X)$ and $V = s(Y)$. The forward AUC is $\Pr(U > V) + \frac{1}{2}\Pr(U = V)$; the reverse AUC is $\Pr(V > U) + \frac{1}{2}\Pr(U = V)$. The events $U > V$, $V > U$, and $U = V$ partition the sample space, so the sum is 1. The same argument holds for pairwise comparisons on the same finite samples.

Equation (10) implies that including both directions of every model pair with equal weight gives a mean AUROC of exactly 50% when both candidates use the same score ranking. Candidate-specific additive calibration offsets leave AUROC unchanged. The shared-autoencoder groups are specified in Appendix A.2. For each pair, the two empirical AUROCs sum to exactly 1 in all three methods. Individual AUROCs range from 45.36% to 54.64% for AEDR, 38.70% to 61.30% for AEROBLADE, and 35.17% to 64.83% for LatentTracer. Thus, a ranking that favors one source reverses its attribution performance when that source becomes the negative class. The 50% group mean follows from averaging these complementary directions, while the shared reconstruction map supplies the same response to both candidates.

B.2 Reconstruction error under CMR

Let $F_2 = \phi \circ D_2$, where ϕ is the fixed feature map in Equation (4). Suppose that F_2 is L_2-Lipschitz on a neighborhood $\mathcal{U}$ containing z_0 and $\widehat{z}_2(x')$. Thus, for all $z, z' \in \mathcal{U}$,

$$\|F_2(z) - F_2(z')\|_2 \le L_2 \|z - z'\|_2. \tag{11}$$

Proof of Equation (5). For the continuous decoder output $x' = D_2(z_0)$, Equation (4) gives

$$\begin{aligned} \sqrt{e_2(x')} &= \|\phi(x') - \phi(D_2(\widehat{z}_2(x')))\|_2 \\ &= \|F_2(z_0) - F_2(\widehat{z}_2(x'))\|_2 \\ &\le L_2 \|z_0 - \widehat{z}_2(x')\|_2 . \end{aligned} \tag{12}$$

The final inequality follows from Equation (11). Both sides are nonnegative, so squaring establishes Equation (5).

Output quantization. For stored images, write $x' = Q(D_2(z_0))$, where Q is the output conversion used to save the image. Define its discrepancy in the same feature space as

$$\varepsilon_{\mathrm{out}} = \|\phi(x') - F_2(z_0)\|_2 . \tag{13}$$

Using Equation (13) and the same local Lipschitz condition, the triangle inequality yields

$$\begin{aligned} \sqrt{e_2(x')} &= \|\phi(x') - F_2(\widehat{z}_2(x'))\|_2 \\ &\le \|\phi(x') - F_2(z_0)\|_2 + \|F_2(z_0) - F_2(\widehat{z}_2(x'))\|_2 \\ &\le \varepsilon_{\mathrm{out}} + L_2 \|\widehat{z}_2(x') - z_0\|_2 . \end{aligned} \tag{14}$$

Squaring both sides gives

$$e_2(x') \le \left(\varepsilon_{\mathrm{out}} + L_2 \|\widehat{z}_2(x') - z_0\|_2\right)^2 . \tag{15}$$

In Equation (15), the identity output conversion has $\varepsilon_{\mathrm{out}} = 0$ and recovers the continuous-output bound.

Reconstruction distances and latent estimates. Pixel mean squared error corresponds to $\phi(a) = a/\sqrt{n}$, where n is the number of scalar image values. Weighted squared differences of fixed image features also have the form in Equation (4) when the weights are nonnegative; their square roots and spatial normalization are absorbed into ϕ. For latent optimization, $\widehat{z}_2(x')$ denotes the code associated with the reported reconstruction error. For stochastic latent estimates, the bound applies to each realization in $\mathcal{U}$.

B.3 Local behavior of symmetric perturbations

Proposition 2 (First-order cancellation) *Fix observer M_k and noise level t. Suppose that $f_{k,t}$ is twice continuously differentiable on an open convex neighborhood $\mathcal{U}$ of $z_t^{(k)}$ and that $\|D^2 f_{k,t}(w)[a,b]\|_2 \le H_{k,t}\|a\|_2\|b\|_2$ for every $w \in \mathcal{U}$ and latent directions a, b. For any $\delta_{t,q}^{(k)}$ with $z_t^{(k)} \pm \delta_{t,q}^{(k)} \in \mathcal{U}$, the average prediction in Equation (7) satisfies*

$$\left\| \bar{z}_{t,q}^{(k)} - f_{k,t}(z_t^{(k)}) \right\|_2 \le \frac{H_{k,t}}{2} \left\| \delta_{t,q}^{(k)} \right\|_2^2. \tag{16}$$

Proof of Proposition 2. For this proof, write $f = f_{k,t}$, $z = z_t^{(k)}$, and $\delta = \delta_{t,q}^{(k)}$. Convexity ensures that both segments $z \pm \lambda\delta$, $0 \le \lambda \le 1$, lie in $\mathcal{U}$. Taylor's formula with integral remainder gives

$$\begin{aligned} f(z+\delta) &= f(z) + Df(z)\delta + R_+, \\ f(z-\delta) &= f(z) - Df(z)\delta + R_-, \\ R_\pm &= \int_0^1 (1-\lambda) D^2 f(z \pm \lambda\delta)[\delta,\delta]\, d\lambda. \end{aligned} \tag{17}$$

The derivative bound gives $\|R_\pm\|_2 \le H_{k,t}\|\delta\|_2^2/2$. Averaging cancels $Df(z)\delta$ and $-Df(z)\delta$, so

$$\left\| \frac{f(z+\delta) + f(z-\delta)}{2} - f(z) \right\|_2 \le \frac{\|R_+\|_2 + \|R_-\|_2}{2} \le \frac{H_{k,t}}{2}\|\delta\|_2^2. \tag{18}$$

Substituting the original notation establishes Equation (16).

The bound controls the local change in the averaged prediction caused by the added perturbation.

C Implementation details

C.1 CMR and reconstruction error-based baselines

CMR performs one encoding–decoding pass through the attack model's autoencoder, as specified in Equation (2). For KL autoencoders, the encoder supplies the posterior mode to the decoder. LatentTracer starts from the candidate encoder's representation and uses 50 latent optimization steps.

C.2 READ inference

Algorithm 1 summarizes source attribution using the fixed reference models, normalizer, classifier, and decision threshold.

C.3 READ response extraction

Before denoising, the deterministic encoder output is converted to the model's latent scale, including any configured shift.

Native noise schedules. For variance-preserving models, the coefficients in Equation (6) are $\alpha_{k,t} = \sqrt{\bar{\alpha}_t^{(k)}}$ and $\sigma_{k,t} = \sqrt{1 - \bar{\alpha}_t^{(k)}}$, where $\bar{\alpha}_t^{(k)}$ is the cumulative signal-retention factor of the native scheduler (Ho et al., 2020). For flow-matching models, $\alpha_{k,t} = 1 - \sigma_{k,t}$, with $\sigma_{k,t}$ supplied by the native flow scheduler (Esser et al., 2024). The three timesteps in $\mathcal{T}_k$ are those closest to target log-SNRs $\{4, 0, -4\}$, using $\log(\alpha_{k,t}^2/\sigma_{k,t}^2)$ for both families.

Algorithm 1 READ: Responses Evidence for Attribution from Denoisers

Require: Image x, candidate M_j, fixed $\mathcal{O}_j, g_j, h_j, \Psi_j, \tau_j$.
Ensure: Attribution score $s_j(x)$ and acceptance decision $b_j(x)$.

```
for each k ∈ O_j do
    Encode x as z_0^(k) using E_k.
    Form noisy latents {z_t^(k)}_{t∈T_k} using Equation (6).
    Construct paired perturbations ±δ_{t,q}^(k) using Equation (19).
    Evaluate f_{k,t}(z_t^(k) ± δ_{t,q}^(k)) for each noise level t and direction q.
    Compare averaged predictions with z_0^(k) to obtain d^(k)(x) using Equation (7).
end for
Stack T_j(x) ← [d^(k)(x)]_{k∈O_j}.
Normalize T̃_j(x) ← g_j(T_j(x)).
Combine normalized responses and reference differences as v_j(x) ← Ψ_j(T̃_j(x)).
Compute s_j(x) ← h_j(v_j(x)) and b_j(x) ← 𝕀[s_j(x) ≥ τ_j].
return s_j(x), b_j(x).
```

Perturbations and response order. For each image and observer, we draw one Gaussian noise tensor $\epsilon^{(k)}$ and two Rademacher tensors $r_q^{(k)}$, $q \in \{1, 2\}$, whose entries independently take values -1 and $+1$ with equal probability. These draws are reused across the three noise levels, with pseudorandom sequences fixed by extraction group. The perturbations are

$$\delta_{t,q}^{(k)} = \max\{0.01\,\mathrm{RMS}(z_t^{(k)}), 10^{-6}\}\, r_q^{(k)}, \qquad q \in \{1, 2\}, \tag{19}$$

where $\mathrm{RMS}(z) = \sqrt{\mathrm{mean}[z^2]}$. The two signs use the same timestep and conditioning. We order the responses by decreasing target log-SNR and then by perturbation index, giving six coordinates and 12 denoiser forward passes per continuous-latent observer. The numerator and denominator of Equation (7) are computed in float32, with $\varepsilon_{\mathrm{num}} = 10^{-20}$.

Clean-latent predictions. For observer M_k with denoiser parameters θ_k, noise prediction gives

$$f_{k,t}(z) = \frac{z - \sigma_{k,t}\epsilon_{\theta_k}(z, t)}{\alpha_{k,t}}. \tag{20}$$

For variance-preserving velocity prediction, $f_{k,t}(z) = \alpha_{k,t}z - \sigma_{k,t}v_{\theta_k}(z, t)$ (Salimans & Ho, 2022); for direct clean-sample prediction, $f_{k,t}$ is the native model output. For flow matching, $f_{k,t}(z) = z - \sigma_{k,t}v_{\theta_k}(z, t)$. Text conditioning uses empty prompts throughout. KD2.1 uses its native image encoder and denoiser, with fixed text conditioning and an image-prior embedding generated from an empty prompt.

Discrete-model implementation. For the VQDM experiments, inputs are resized to the released model's required 256×256 resolution and encoded as 32×32 discrete tokens. At timesteps $\{10, 50, 90\}$, we draw two perturbed token sequences from the native forward process, compute mean cross-entropy for the original tokens, and divide by the natural logarithm of the vocabulary size. This produces six response coordinates using six native token-denoising passes.

C.4 READ FEATURES, REFERENCE SELECTION, AND CALIBRATION

Unless otherwise stated, READ uses $K = 7$ models, comprising the candidate and six reference models. Calibration uses 500 clean images per registered source, 6,000 in total, separate from the test images. Observer selection starts with the candidate itself. At each step, we tentatively add each remaining model and choose the one with the highest mean balanced accuracy under five-fold stratified cross-validation on calibration data. Selection uses standardized responses, candidate responses, and reference differences with class-balanced logistic regression. This produces nested sets evaluated at $K \in \{1, 3, 5, 7, 9\}$.

We implement normalization and classification with scikit-learn (Pedregosa et al., 2011). For final classification, g_j is a QuantileTransformer with 500 quantiles fitted on calibration data to map

each response coordinate to standard normal quantiles. For each candidate j, $\widetilde{\mathbf{d}}^{(k)}(x)$ is the row of $\widetilde{T}_j(x) = g_j(T_j(x))$ corresponding to observer k. When $K > 1$, Ψ_j concatenates (1) all flattened normalized observer responses; (2) the candidate's own normalized response; (3) coordinate-wise reference-minus-candidate differences; (4) their mean, minimum, maximum, and standard deviation across references; (5) each observer's mean, standard deviation, and last-minus-first coordinate across its six responses; and (6) the candidate's normalized ranks among observers. For $K = 1$, only its own six normalized responses are used.

Fusion variants. Score fusion (SF) fits logistic regression to normalized READ and baseline reconstruction scores. Feature fusion (FF) appends the normalized baseline score to READ features and fits gradient boosting. Both use the same clean calibration images and test images as the standalone methods.

READ and FF use HistGradientBoostingClassifier with 250 maximum iterations, learning rate 0.06, at most 15 leaf nodes, L_2 regularization 1.0, balanced class weights, early stopping, and random seed 0. SF uses logistic regression with $C = 1.0$, balanced class weights, 1,000 maximum iterations, and seed 0. Normalizers and classifiers are fitted on clean calibration data and frozen with the default binary decision boundaries. Fusion normalization for READ and baseline scores also uses only calibration data.

C.5 Additional attribution methods and evaluation protocols

Table 3 uses the following method-specific datasets and protocols. We evaluate Model-Fingerprint (Yang et al., 2025) on the authors' public text-to-image subset, LIDA (Wang et al., 2026b) on GenImage, and FSD (Nguyen et al., 2025) on Synthbuster. Model-Fingerprint matches image fingerprints against a clean reference gallery. LIDA uses its released full-image configuration and retrieves source labels from a clean gallery. FSD uses its released 14-class attribution head.

Reconstruction and source decisions. CMR uses the SD1.5, SD2.1, and SDXL autoencoders for Model-Fingerprint; SD1.4 and SD1.5 for LIDA; and SDXL for FSD. These experiments preserve the input resolution, with reflection padding to multiples of eight and cropping back to the original dimensions after decoding. Tiled encoding and decoding use 512-pixel tiles with 0.25 overlap, and reconstructed images are saved as lossless PNG files. For the additional binary comparisons, both the M_1 positives and M_2 negatives pass through the same attack autoencoder. The M_1 and M_2 acceptance rates are computed on the M_1 positives. LIDA and FSD use candidate-specific thresholds calibrated to a target 5% FPR on separate clean images. Model-Fingerprint uses other-source gallery images for negative calibration. The reference galleries, attribution heads, and thresholds remain fixed during attack testing.

C.6 Control, measurement, and ablation settings

JPEG and Gaussian-blur controls. The comparisons in Table 4 evaluate JPEG compression and Gaussian blur on the same test images as CMR for all 12 attack models. For each attack model, we select each perturbation's strength independently to approximate CMR's development-set LPIPS-Alex distribution. The objective sums the absolute mean difference, absolute median difference, and mean absolute difference over percentile points from the 10th to the 90th. We evaluate JPEG qualities $\{70, 75, 80, 85, 90, 95\}$ and Gaussian blur radii $\{0.25, 0.5, 0.75, 1.0, 1.25\}$ on development images, then retain the minimizing value within each perturbation family. SD3-Medium, SD3.5-Medium, and SD3.5-Large use JPEG quality 75; the remaining models use quality 70. KD2.1 and VQDM use Gaussian blur radius 0.75; the remaining models use radius 0.5. These strengths and all attribution thresholds are fixed before test evaluation. Table 4 reports attribution metrics averaged equally over model pairs. Mean test-set LPIPS is 0.0369 for CMR, 0.0223 for JPEG, and 0.0293 for Gaussian blur, using the same model-pair weighting.

Denoising-response measurements. Figure 6a–b compares images generated by SD2.0 and SD2.1, which share an autoencoder. For each image, the relative response is the normalized SD2.1 recovery error minus the normalized SD2.0 recovery error, averaged over both perturbation directions. Normalization is fitted on clean calibration images and held fixed after CMR. Blue solid

and red dashed lines denote SD2.0 and SD2.1 sources, respectively; error bars show the source mean ± 1.96 standard errors across images. Both source image sets undergo reconstruction with the shared autoencoder in panel (b).

Component ablations. The variants in Table 5 use the fixed reference sets, clean-calibration normalizer, and classifier hyperparameters from Appendix C.4. Low, intermediate, and high noise retain the first, middle, or last two response coordinates, respectively. All variants reuse the cached symmetric predictions, with the perturbations unchanged. The noise-level ablations test which recovery conditions contribute to attribution; the feature ablation tests how explicitly representing differences, summary statistics, and ranks helps the classifier use the same underlying responses.

Response-extraction timing. For each observer, we divide the logged extraction duration by the number of processed images. For a model pair, we sum these costs over the union of observers required for the M_1 and M_2 decisions, counting a shared observer once. The estimates in Figure 6e average these sums equally over model pairs and include initialization overhead recorded in the extraction logs.

D Supplementary experimental results

D.1 Natural-image attribution by autoencoder type

Table 8 expands Table 2 and Figure 5b by attack-autoencoder family. TPR uses the unchanged generated positives; FPR and AUROC reflect reconstruction of the natural negatives under fixed thresholds.

For the KL autoencoders, CMR raises baseline FPR to 77.67–99.97%, indicating that reconstruction often supplies enough evidence for accepting a natural image as a model output. READ keeps FPR at 13.23% for 4-channel models and 1.33% for 16-channel models, while retaining generated-image TPRs of 85.09% and 89.33%. Its low false acceptance therefore accompanies continued recognition of generated positives, consistent with denoising responses retaining information beyond autoencoder reconstruction.

For KD2.1 and VQDM, READ has FPRs of 21.00% and 33.80% and corresponding TPRs of 73.60% and 75.00%. VQDM also differs in its baseline behavior: AEROBLADE and LatentTracer FPRs fall by 4.40 and 4.00 points after CMR, despite remaining high in absolute terms. Their high false acceptance is already present on clean natural images, whereas KD2.1 exhibits sharp attack-induced increases across all three baselines. READ achieves both lower FPR and higher TPR in the evaluated KL families than for KD2.1 or VQDM.

D.2 Image fidelity and example selection

Table 9 separates image fidelity by attack model for generated and natural inputs, revealing variation within the means reported in Tables 1 and 2. Natural inputs are prepared at 1024×1024 for SDXL-base and SD3.5-Large, 256×256 for VQDM, and 512×512 for the remaining models; each output is compared with the input at its evaluation resolution. Distortion varies across attack models: the SD3 family achieves generated-image LPIPS of 0.0155–0.0156 and DINO similarity of 0.9947–0.9948, whereas VQDM has the highest LPIPS on both generated and natural images, at 0.0738 and 0.0952. The SD3 family also preserves natural inputs closely, yet AEDR and LatentTracer accept about 98% of these reconstructions (Table 8). This combination of high fidelity and high false acceptance shows that severe visual degradation is unnecessary for reconstruction to be mistaken for generation.

Figure 4 uses one SD1.5 image for all 11 attack models with $M_2 \neq M_1$ and one natural image for all 12 attack models. For each attack model, we compute the median and standard deviation of LPIPS over the examples available for every displayed model. We divide each example's absolute deviation from the median by this standard deviation and average these deviations over the displayed attack models. The generated example minimizes this average. For the natural example, we inspect the 16 images with the smallest average deviations and select a landscape photograph to show scene structure and texture; the selected image ranks ninth by this criterion. These selection criteria are independent of attribution scores and decisions.

Table 8: Natural-image attribution after CMR Attack, grouped by attack autoencoder (%). Parentheses report signed changes from Clean in percentage points. FPR measures acceptance of reconstructed natural images by M_2. TPR uses unchanged clean images generated by M_2, and AUROC compares these positives with reconstructed natural negatives. Thresholds remain fixed, and candidate models receive equal weight within each group.

Method	TPR (clean generated)	CMR Attack	
		FPR	AUROC
4-channel KL autoencoders			
AEDR	99.40 (+0.00)	99.97 (+80.11)	39.88 (−50.24)
AEROBLADE	96.83 (+0.00)	90.80 (+80.60)	43.68 (−52.58)
LatentTracer	92.14 (+0.00)	94.57 (+75.43)	29.45 (−60.99)
READ	85.09 (+0.00)	13.23 (−0.06)	92.15 (+0.18)
16-channel KL autoencoders			
AEDR	95.33 (+0.00)	98.33 (+66.66)	17.70 (−60.42)
AEROBLADE	97.33 (+0.00)	77.67 (+49.67)	70.01 (−17.52)
LatentTracer	92.67 (+0.00)	98.00 (+65.67)	22.56 (−54.95)
READ	89.33 (+0.00)	1.33 (−1.34)	98.68 (+0.42)
4-channel VQModel (KD2.1)			
AEDR	93.20 (+0.00)	100.00 (+96.80)	0.72 (−98.23)
AEROBLADE	97.00 (+0.00)	93.60 (+93.00)	66.22 (−33.69)
LatentTracer	94.00 (+0.00)	99.00 (+99.00)	19.13 (−80.79)
READ	73.60 (+0.00)	21.00 (+0.60)	83.13 (−0.91)
256-channel VQModel (VQDM)			
AEDR	92.20 (+0.00)	91.80 (+13.80)	54.94 (−11.21)
AEROBLADE	92.20 (+0.00)	65.40 (−4.40)	74.14 (+1.95)
LatentTracer	98.00 (+0.00)	88.00 (−4.00)	65.19 (+7.84)
READ	75.00 (+0.00)	33.80 (−4.40)	77.94 (+1.61)

Table 9: Image fidelity after CMR for the generated-image and natural-image evaluations in Tables 1 and 2. For each M_2, generated-image metrics are averaged within each M_1–M_2 pair and then equally across $M_1 \neq M_2$; natural-image metrics are averaged over its reconstructed natural inputs. Mean gives equal weight to each M_2. PSNR is in dB, and DINO denotes DINOv2 cosine similarity.

Attack model M_2	Generated images				Natural images			
	PSNR	SSIM	LPIPS	DINO	PSNR	SSIM	LPIPS	DINO
SD1.5	27.79	0.8196	0.0464	0.9817	26.63	0.7450	0.0808	0.9689
SD2.0	30.10	0.8648	0.0360	0.9871	27.71	0.7694	0.0740	0.9771
SD2.0-base	30.06	0.8651	0.0357	0.9870	27.71	0.7694	0.0740	0.9771
SD2.1	30.05	0.8647	0.0361	0.9871	27.71	0.7694	0.0740	0.9771
SD2.1-base	30.02	0.8645	0.0358	0.9870	27.71	0.7694	0.0740	0.9771
SDXL-base	28.95	0.8322	0.0412	0.9856	33.23	0.8926	0.0567	0.9929
SDXL-Turbo	29.22	0.8285	0.0426	0.9858	28.07	0.7860	0.0731	0.9781
SD3-Medium	33.34	0.9086	0.0155	0.9947	32.78	0.9118	0.0257	0.9941
SD3.5-Medium	33.22	0.9069	0.0155	0.9947	32.71	0.9113	0.0257	0.9940
SD3.5-Large	33.25	0.9071	0.0156	0.9948	37.74	0.9684	0.0125	0.9986
KD2.1	28.14	0.8102	0.0488	0.9798	27.68	0.7756	0.0796	0.9654
VQDM	25.99	0.7916	0.0738	0.9687	23.40	0.6739	0.0952	0.8974
Mean	30.01	0.8553	0.0369	0.9862	29.42	0.8119	0.0621	0.9748

Pixel differences after CMR. Figures 7 and 8 compare three SD1.5 images and three natural images with their CMR reconstructions under SD2.0 and SD3.5-Medium. Each row uses the same original image for both attack models. The differences are computed at the original 512×512 evaluation resolution. All heatmaps use a common linear color scale from 0 to 0.15; values above 0.15 receive the maximum color.

We select examples near the lower quartile, median, and upper quartile of LPIPS averaged over the two attack models, after dividing each model's LPIPS values by its median. Natural-image examples are scene photographs near these ranks. The maps reveal changes in edges and fine textures, including buildings, vegetation, and waves, while the reconstructed images preserve the scene layout. SD3.5-Medium produces smaller pixel differences than SD2.0 in these examples, consistent with its higher PSNR.

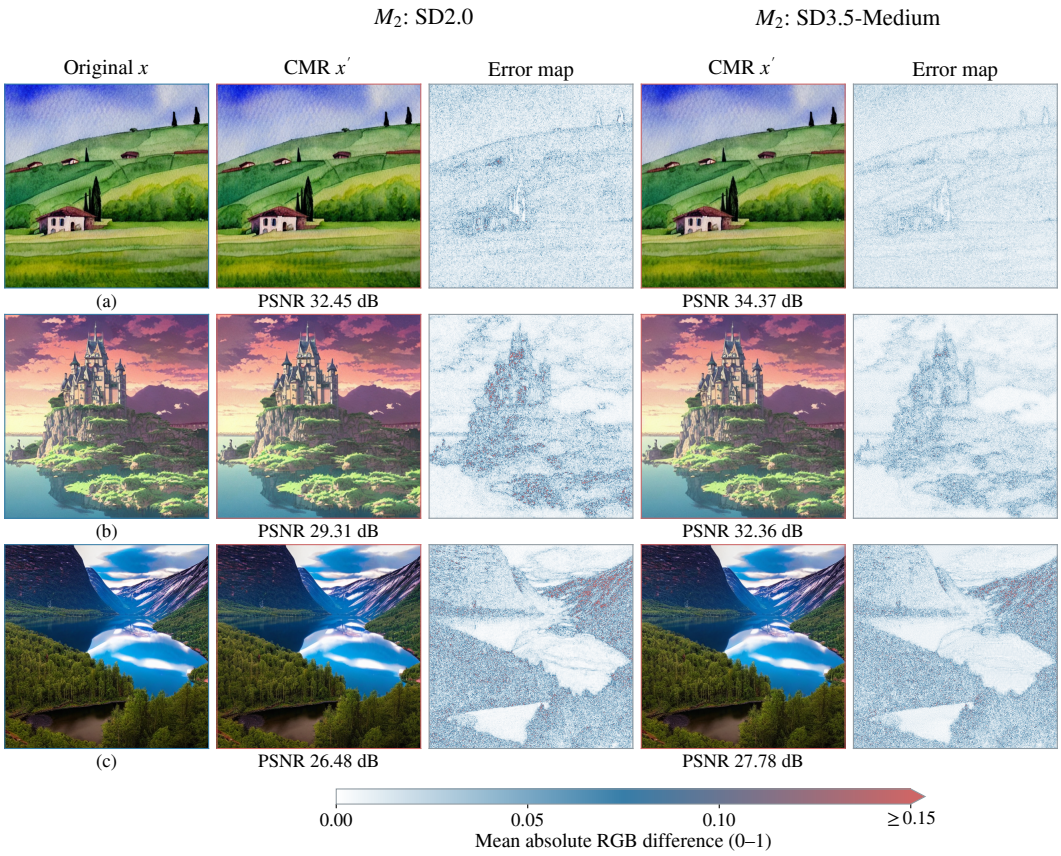


Figure 7: Pixel differences after CMR on generated images. Rows (a–c) show three images generated by M_1 = SD1.5. Each original is followed by its reconstruction under SD2.0 and SD3.5-Medium and the corresponding mean absolute RGB difference. PSNR is measured against the original image. The heatmaps share the same linear scale, with values above 0.15 saturated.

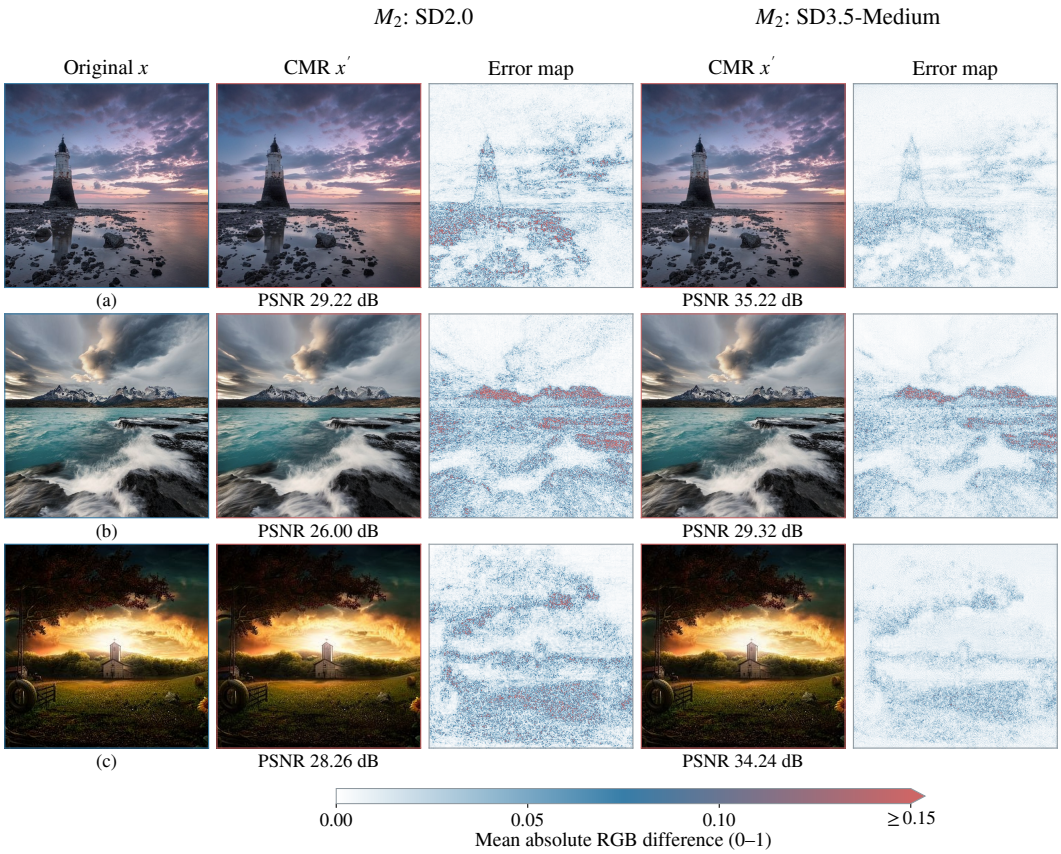


Figure 8: Pixel differences after CMR on natural images. Rows (a–c) show three scene photographs reconstructed by SD2.0 and SD3.5-Medium. Each error map gives the mean absolute RGB difference between the CMR output and its original image. PSNR and the shared linear color scale follow Figure 7.

D.3 Score distributions under CMR

Figures 9 and 10 extend Figure 3 to AEDR and LatentTracer using the same SD1.5 input images and attack models. The horizontal coordinate is $s_j(x) - \tau_j$ for Clean and $s_j(x') - \tau_j$ after CMR; nonnegative margins indicate acceptance. The symmetric logarithmic axes retain the long negative tails while resolving scores near the threshold. The linear regions span $[-0.1, 0.1]$ for AEDR and $[-10^{-5}, 10^{-5}]$ for LatentTracer. Each histogram uses 50 equal intervals in the transformed coordinate, with frequencies normalized separately for each distribution.

For both methods, CMR shifts the attack-model scores from predominantly negative margins into the acceptance region. The effect therefore extends to both AEDR's successive-error ratio and LatentTracer's optimized latent reconstruction, despite their different scoring procedures. This behavior is consistent with CMR supplying a latent code that the attack-model decoder can reproduce, making the reconstructed image easier to explain through that decoder. LatentTracer's accepted scores cluster closer to the threshold for SD3 and SD3.5 than for SD2, although all four attack-model distributions shift toward acceptance.

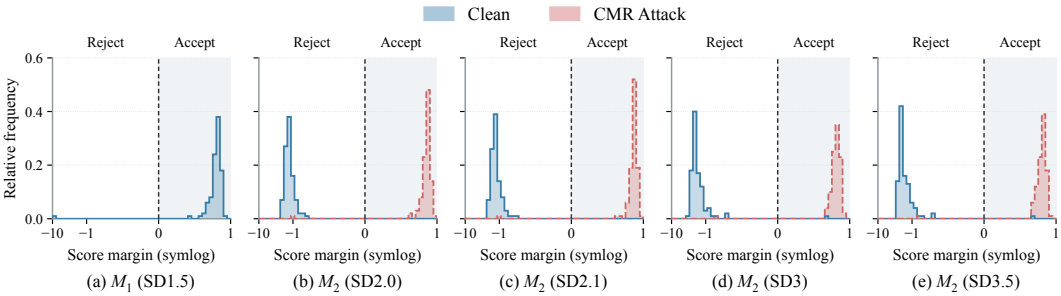


Figure 9: AEDR score distributions before and after CMR on SD1.5 images. Panel (a) shows Clean scores under M_1; (b–e) compare Clean and CMR Attack scores under SD2.0, SD2.1, SD3-Medium, and SD3.5-Medium, respectively. Shading marks acceptance.

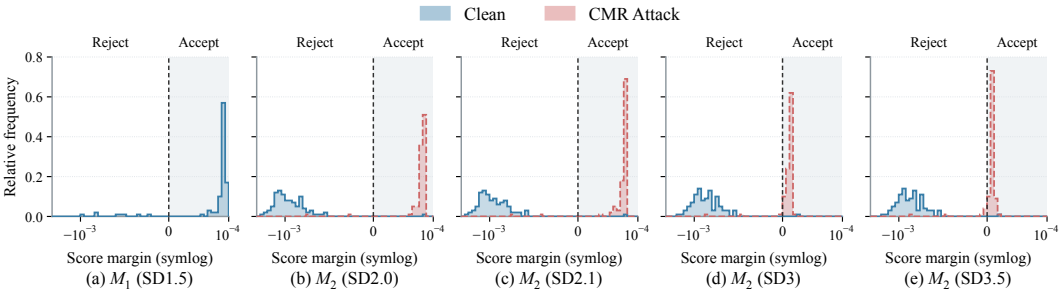


Figure 10: LatentTracer score distributions before and after CMR on SD1.5 images. Panel (a) shows Clean scores under M_1; (b–e) compare Clean and CMR Attack scores under SD2.0, SD2.1, SD3-Medium, and SD3.5-Medium, respectively. Shading marks acceptance.

D.4 Attack-model selection and further processing

These experiments test whether an attacker can increase strict reattribution success against READ by choosing the attack model, repeating reconstruction, or selecting additional image transformations. Table 10 reports strict RTASR under these three settings, and Table 11 measures cumulative distortion relative to the original images. A0 selects among evaluated attack models for each original image; A1 and A2 retain fixed source–attack model combinations. A0 uses the observed attribution decisions, and A2 uses attribution scores to select outputs, whereas the single-pass CMR attack requires no attribution queries.

A0: attack-model selection. For original sample entry i, let $\mathcal{T}_i$ contain the indices of the evaluated attack models and $q_{i,j}$ be the strict-success indicator for attack model M_j. A0 reports

$$\mathrm{RTASR}_{A0} = \frac{1}{N_{\mathrm{orig}}} \sum_{i=1}^{N_{\mathrm{orig}}} \mathbb{I}\left[\max_{j \in \mathcal{T}_i} q_{i,j} = 1\right]. \tag{21}$$

Here N_{orig} is the number of evaluated original images, and $\mathcal{T}_i$ contains the attack models evaluated for image i. Each original image contributes once.

A1: repeated autoencoder reconstruction. For fixed source–attack model combinations, we apply $x^{(\ell)} = A_2(x^{(\ell-1)})$ with $x^{(0)} = x$. Every pass uses the same frozen attribution rules. Quality is always measured against the original image, so it reflects cumulative distortion rather than only the latest increment.

A2: quality-constrained transformation selection. After one autoencoder reconstruction pass, the candidates are identity, JPEG quality 90, JPEG quality 70, resizing to 448×448 and bicubic

upsampling to 512×512, Gaussian blur with $\sigma = 0.5$, Gaussian noise with standard deviation 1 in pixel units, and a combination of JPEG 90, resizing, blur, and noise.

Candidates must satisfy PSNR$\geq$ 30, SSIM$\geq$ 0.9, LPIPS$\leq$ 0.1, and DINO$\geq$ 0.95 relative to the single-pass reconstructed image. For each method, we select the eligible output with the largest attack-model acceptance margin minus original-model acceptance margin. Methods may select different transformations. If none qualifies, we retain the identity output. Attribution models and thresholds are not retrained. Quality filtering is relative to the single-pass reconstructed image; the final quality reported below is relative to the original.

Table 10: Strict RTASR (%) under adaptive evaluation. A0 measures success per original image after attack-model selection. A1 and A2 evaluate fixed image–model combinations after repeated reconstruction or selected post-processing.

Setting	READ	AEDR + FF	AEROBLADE + FF
A0 Per-image attack-model selection	24.75	70.96	62.25
A1 One pass	4.00	16.04	9.12
A1 Two passes	3.92	16.54	9.25
A1 Three passes	4.46	17.50	10.50
A2 Quality-constrained selection	14.54	29.75	22.00

Table 11: Cumulative visual distortion after repeated autoencoder reconstruction and quality-constrained selection. The two A2 rows use outputs selected separately for each fusion method.

Condition	PSNR ↑	SSIM ↑	LPIPS ↓	DINO ↑
A1 One pass	30.729	0.8774	0.0334	0.9867
A1 Two passes	28.638	0.8464	0.0558	0.9767
A1 Three passes	27.109	0.8152	0.0848	0.9617
A2: AEDR + FF	30.321	0.8696	0.0474	0.9850
A2: AEROBLADE + FF	30.353	0.8698	0.0468	0.9849

READ has the lowest strict RTASR among the evaluated configurations under all three settings (Table 10). With per-image attack-model selection, READ's 24.75% strict RTASR is below the 62.25–70.96% of the fusion methods. Selection can exploit different attack models on different images, but READ remains less susceptible than variants that incorporate reconstruction evidence.

Repeating reconstruction from one to three passes changes READ's strict RTASR only from 4.00% to 4.46%, while LPIPS rises from 0.0334 to 0.0848 (Table 11). Success also changes modestly for the fusion methods. Further passes through the same autoencoder incur greater distortion with little gain in strict reattribution success. By contrast, score-guided transformation selection raises READ's strict RTASR to 14.54% on the fixed model combinations, indicating greater sensitivity to selected post-processing than to repeated reconstruction alone.

After selected post-processing, only-M_1 acceptance is 20.58% for READ, compared with 7.50% and 11.71% for AEDR + FF and AEROBLADE + FF. READ rejects both candidates on 43.92% of images. Its lower strict reattribution rate thus accompanies both more frequent correct exclusive attribution and rejection of the tested candidates.

D.5 Joint source decisions and recovery

Table 12 partitions post-CMR decisions into the four outcomes defined in Appendix A.4.

For a baseline B, let q_i^B indicate a transition from only M_1 before attack to only M_2 after attack. The recovery rate of a defense D on this fixed subset is

$$\text{Recovery}(D \mid B) = \frac{\sum_i q_i^B \, b_1^D(x_i')[1 - b_2^D(x_i')]}{\sum_i q_i^B}. \tag{22}$$

Every method is evaluated on the same baseline-specific failure subset. The confidence intervals in Table 12 use 2,000 bootstrap replicates clustered by original-image hash, with each image's repeated model-pair evaluations retained together. They condition on the fitted classifiers and evaluated model set.

Table 12: Joint source decisions and READ recovery after CMR (%). Joint rates are model-pair means over all attacked images. Recovery evaluates READ on each reconstruction baseline's successful-attack subset and requires accepting only M_1.

Method	All attacked images				Baseline failure subset	
	Only M_1	Only M_2	Both	Neither	READ recovery	95% interval
AEDR	0.07	53.08	46.79	0.06	46.53	[45.34, 47.80]
AEROBLADE	0.65	45.57	48.02	5.75	45.03	[43.89, 46.21]
LatentTracer	0.47	54.20	44.06	1.27	45.34	[42.90, 47.71]
READ	48.12	7.48	6.12	38.27	–	–

The reconstruction baselines predominantly accept either M_2 alone or both candidates, combining false attribution with source ambiguity. At most 0.65% of images receive only-M_1 acceptance, compared with 48.12% for READ. READ reduces both only-M_2 and joint acceptance, shifting decisions toward correct exclusive attribution to M_1. READ rejects both candidates on 38.27% of images.

On the fixed subsets where CMR changes a baseline decision from only M_1 to only M_2, READ restores only-M_1 acceptance for 45.03–46.53% of images. Recovery across all three baselines indicates that denoising responses retain usable original-source evidence after reconstruction has redirected the baselines' decisions.