COMPOSITIONAL ADVERSARIAL TRAINING FOR ROBUST VISUAL WATERMARKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Robust watermarking is typically trained with random post-processing augmentation, but random sampling under-covers the combinatorial space of realistic attack pipelines and rarely encounters the rare compositions that actually break detection. This leads to unstable training and poor sample efficiency. We instead formulate watermark robustness as a min-max problem over a structured space of compositional transformations. We propose Compositional Adversarial Training (CAT), a plug-in framework that learns a sequential differentiable adversary that observes the current intermediate image and adaptively selects an attack family at each step to maximally disrupt message recovery. CAT combines a straight-through Gumbel-Softmax attack selection with entropy regularization, allowing the backward pass to be end-to-end differentiable and aggregate gradient information across attack families, yielding faster, smoother convergence without collapsing to a single attack mode. We evaluate CAT on post-generation watermarks VideoSeal 0.0, VideoSeal 1.0, and PixelSeal and in-generation WMAR under both single-step and two-step attack suites, on in-distribution and multiple out-of-distribution image and video benchmarks. CAT consistently outperforms random-augmentation baselines trained for the same number of gradient updates and the same augmentation budget, with the largest gains on hard composed attacks and OOD evaluations; in the SA-1B and CLIC image benchmarks, it improves overall watermark capacity by up to 74.3% in the single-step setting and 24.2% in the compositional setting. In the autoregressive setting, CAT raises difficult geometric TPR@FPR$= 1\%$ by up to 18 percentage points. These results show that robust visual watermarking benefits from training against adaptive compositional adversaries rather than independent random corruptions.

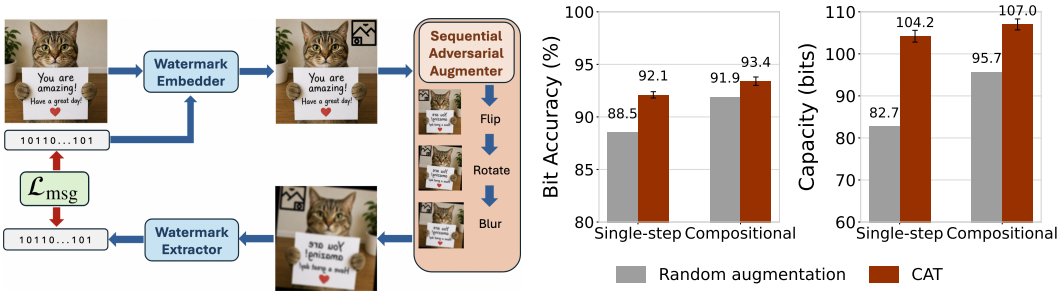


Figure 1: Overview of Compositional Adversarial Training (CAT) for visual watermarking. The result panel reports mean overall bit accuracy and capacity across VideoSeal 0.0, VideoSeal 1.0, and PixelSeal on SA-1B and CLIC. On average, CAT improves overall watermark capacity by 26.0% under single-step attacks and 11.7% under compositional attacks.

1 INTRODUCTION

As synthetic and redistributed visual media proliferate, visual watermarking is increasingly used for provenance and misuse detection. In practice, watermarked images are rarely observed in their clean

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.

form and instead undergo recompression, resizing, cropping, blur, color shifts, and compositions of these operations (Jia et al., 2021; An et al., 2024; Ding et al., 2024). Robustness therefore depends on surviving structured post-processing pipelines rather than isolated perturbations.

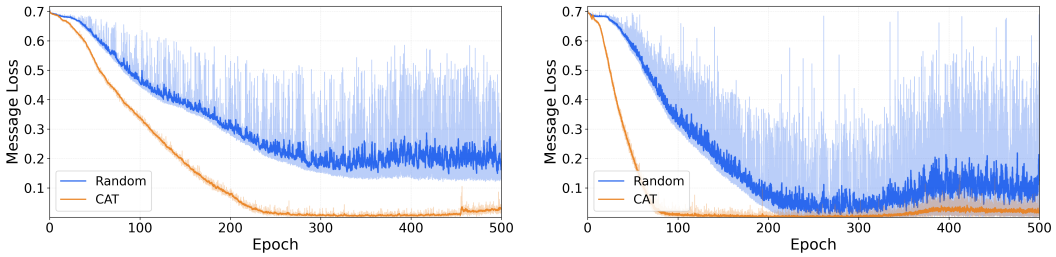
Current training relies primarily on random augmentation, which samples individual corruptions and parameters independently (Jovanović et al., 2025; Souček et al., 2025). While this improves performance under common distortions, it poorly reflects deployment conditions, where attacks are compositional, order-dependent, and highly imbalanced. As a result, training tends to overemphasize easy transformations while under-sampling rare but critical failure cases. This leaves random augmentation under-targeted and inefficient, failing to consistently surface the hard, order-sensitive failures that dominate real-world degradation, as shown in Figure 2. On the other hand, pure adversarial training can improve coverage, but it is unstable in this setting: an unconstrained inner maximization may rapidly converge to destructive transformations that remove the watermark entirely and collapse the learning signal (Wen & Aydore, 2019). Related approaches such as RoMark, GAT, and Adversarial AutoAugment also require repeated decoder evaluations, inner optimization steps, or multiple policy-augmented passes, making them difficult to apply to watermark training at scale (Wen & Aydore, 2019; Hsiung et al., 2023; Zhang et al., 2019b).

We propose Compositional Adversarial Training (CAT), which adaptively selects and orders attacks in a shared single-pass update. Rather than directly optimizing over the space of complete attack sequences, CAT factorizes attack selection into sequential decisions, each conditioned on the intermediate image produced by preceding transformations. This recurrence allows the adversary to respond to both the watermark model's evolving weaknesses and the realized effects of earlier attacks. Straight-through Gumbel-Softmax enables end-to-end attack selection, while entropy regularization maintains policy diversity and prevents premature collapse. CAT preserves inference behavior and adds only 20–30% training overhead.

Empirically, CAT improves robustness across VideoSeal 0.0, VideoSeal 1.0 (Fernandez et al., 2024), and PixelSeal (Souček et al., 2025) under both in-distribution and distribution-shifted evaluations. The gains are most pronounced under difficult geometric and compositional attacks, with capacity improvements of up to **74.3%** under single-step attacks and **24.2%** under composed attacks, while also converging faster than random augmentation. Controlled ablations further isolate the source of these gains: direct prediction of complete attack combinations collapses, while non-image-conditioned and alternative adaptive strategies underperform CAT, supporting the importance of sequential, state-conditioned attack selection. The same framework extends to autoregressive watermarking, where CAT raises geometric TPR@FPR$= 1\%$ by up to 18 percentage points. Figure 1 summarizes the setting and key results.

Our contributions are summarized as follows:

- **Problem formulation.** We identify a core limitation of random augmentation for watermark robustness: uniform sampling can over-emphasize easy transformations while under-sampling rare but difficult compositional failures, resulting in brittle robustness and inefficient training.
- **Method.** We introduce CAT, which factorizes compositional attack selection into sequential decisions conditioned on the evolving attacked image, allowing each attack to adapt to preceding effects without direct optimization over complete sequences.
- **Optimization.** We show how to train CAT end-to-end using a straight-through Gumbel-Softmax controller, entropy regularization, and a shared single-pass update, enabling adaptive attack generation with weakness-aligned learning signals while avoiding policy collapse, keeping training practical, and leaving inference unchanged.
- **Empirical evidence.** We demonstrate gains across multiple watermarking backbones, attack horizons, out-of-distribution benchmarks, and zero-shot long-chain simulated deployment pipelines, and show through controlled ablations that sequential factorization and intermediate-state conditioning outperform direct composition prediction and alternative adaptive attack-selection strategies.



(a) Single-step augmentation training.

(b) Compositional augmentation training.

Figure 2: **Random augmentation creates unstable training due to inefficient augmentation allocations, whereas the learned adversary consistently targets the model's current weaknesses.** Panel (a) shows the single-step setting and panel (b) the compositional setting. Under random augmentation, most sampled attacks are already easy and contribute little, while occasional hard or order-sensitive samples arrive sporadically, causing training to oscillate between uninformative updates and abrupt failures. CAT reduces this instability by adaptively concentrating on informative vulnerabilities, especially in the compositional regime where attack order matters.

2 METHOD

Overview. Given a watermarked image, the adversary chooses a short sequence of attacks conditioned on the current intermediate image, relaxes discrete operation choice with a straight-through Gumbel-Softmax estimator, and regularizes the attack policy with entropy. The attack depth T controls the length of the composition: $T = 1$ corresponds to a single attack, while $T = 2$ enables two-step compositions such as blur followed by compression, and larger values extend the same sequential construction to longer attack chains. The main experiments use $T = 2$ as the primary compositional training setting. Appendix E evaluates transfer to three-step chains without retraining and also reports native three-step training, showing that the formulation applies beyond the two-step setting. Because the adversary is differentiable and short-horizon, the inner problem can be optimized in the same training pass without an expensive search over long attack sequences; this shared-pass property is what keeps adaptive compositional training practical at watermark scale. Figure 3 gives a conceptual overview of the full training pipeline.

2.1 PROBLEM FORMULATION AND SEQUENTIAL ADVERSARIAL AUGMENTATION

The goal of robust visual watermarking is to embed a hidden message into an image while preserving visual fidelity and enabling reliable decoding after post-processing. Let $x \in \mathbb{R}^{H\times W\times C}$ denote a clean image and $m \in \{0,1\}^{d_m}$ a binary message of length d_m. The watermarking system consists of an embedder E_θ and an extractor D_ψ. The embedder produces a watermarked image $x_{\text{wm}} = E_\theta(x, m)$, which must satisfy an imperceptibility constraint, typically enforced via a perceptual similarity metric $\mathcal{L}_{\text{perc}}(x, x_{\text{wm}}) \le \epsilon$. After embedding, the image is subjected to a post-processing attack $t \in \mathcal{T}$, where $\mathcal{T}$ is a bounded but expressive set of realistic transformations. Crucially, $\mathcal{T}$ is compositional: an attack may consist of multiple parameterized operations applied in sequence, such as resizing, blurring, and compression in different orders. The extractor receives the attacked image $\tilde{x} = t(x_{\text{wm}})$ and predicts the recovered message $\hat{m} = D_\psi(\tilde{x})$. We therefore seek parameters (θ, ψ) that minimize message recovery error under adversarially selected attacks from $\mathcal{T}$. This yields the min–max objective

$$\min_{\theta,\psi} \max_{\phi} \; \mathbb{E}_{x,m}\Big[\mathcal{L}_{\text{msg}}\big(D_\psi(\mathcal{A}_\phi(E_\theta(x,m))),\, m\big)\Big], \tag{1}$$

where $\mathcal{A}_\phi$ is a learned adversary that parameterizes the compositional attack space $\mathcal{T}$. The inner maximization ranges over attack identities and their order, subject to bounded realistic ranges; continuous transformation parameters are sampled from their prescribed distributions rather than maximized directly. Unlike conventional adversarial training based on small ℓ_p-bounded perturbations, this objective captures the structured, order-sensitive nature of real-world image corruptions (An et al., 2024; Ding et al., 2024). It encourages robustness to realistic post-processing pipelines rather than to isolated perturbations. In practice, we evaluate robustness using two primary metrics: (i) bit

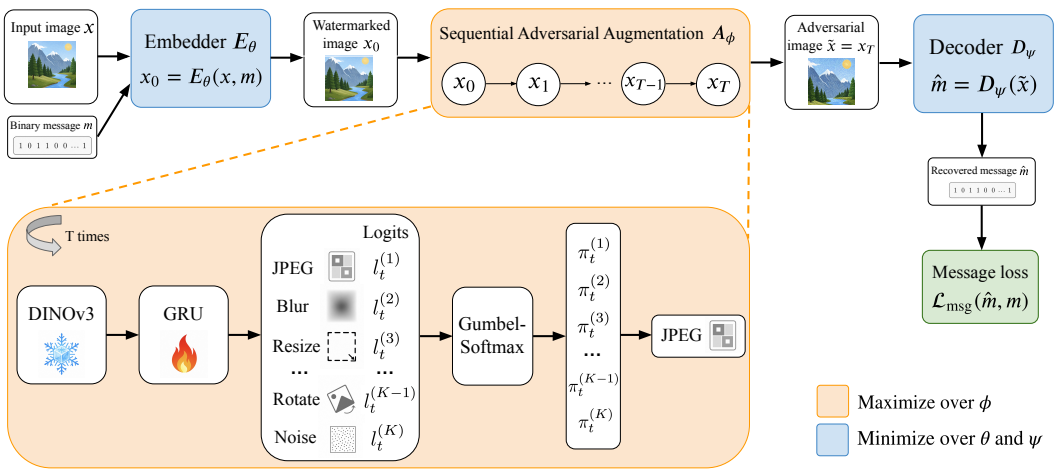


Figure 3: Conceptual overview of the proposed training pipeline. The embedder writes message m into image x to produce the watermarked image x_0. The adversary then repeatedly observes the current image x_t, uses a recurrent controller to produce logits, selects an attack family via straight-through Gumbel-Softmax, and applies differentiable attacks to obtain x_{t+1}. After T steps, the final attacked image x_T is passed to the extractor, and training optimizes message recovery loss. Entropy regularization, described in Section 2.2, keeps the attack policy diverse rather than collapsing to a single destructive sequence.

accuracy, the fraction of correctly recovered bits, and (ii) usable capacity, the number of bits that can be reliably transmitted under attack.

Sequential attack generator. We instantiate $\mathcal{A}_\phi$ as a controller over a library of K differentiable attack primitives $\{f_i\}_{i=1}^K$. Starting from the watermarked image $x_0 = E_\theta(x, m)$, the adversary applies

$$x_{t+1} = f_{a_t}(x_t, \lambda_t), \quad t = 0, \ldots, T-1, \tag{2}$$

where $a_t \in [K] := \{1, \ldots, K\}$ selects the attack primitive and $\lambda_t \in \Lambda_{a_t}$ denotes its continuous parameters, with each Λ_i defining a bounded valid range for attack family i. The primitive library is built from standard differentiable post-processing operations used in robust watermarking, covering major corruption families—including value distortions, compression artifacts, and geometric transformations—while remaining visually plausible and efficiently composable during training. The final attacked image is $\tilde{x} = x_T$. Unlike a one-shot augmentation layer, this sequential generator lets later operations depend on the intermediate image produced by earlier ones, which is essential for order-sensitive compositions. In particular, depth $T = 1$ recovers a single learned attack, while depth $T = 2$ enables two-step attack compositions; increasing T extends the controller to longer sequences.

Sampling transformation parameters. We intentionally do not adversarially optimize transformation parameters. Direct parameter maximization rapidly drives continuous transformations toward maximally destructive boundary values, resulting in degenerate images and collapse of the watermark learning signal. Sampling parameters instead exposes the watermark model to the full prescribed severity distribution while allowing the controller to focus its adversarial capacity on selecting which transformations to apply and in which order. This also means that CAT does not require differentiability with respect to transformation parameters, enabling augmentation families with non-differentiable parameterizations.

Differentiable attack selection. Optimizing equation 1 requires gradients through the choice of a_t. Directly sampling a discrete operation would block backpropagation, and score-function estimators are high-variance because the loss depends on an entire composed attack sequence. We therefore use a straight-through Gumbel-Softmax relaxation (Jang et al., 2017). At each step t, a controller produces logits $\ell_t \in \mathbb{R}^K$ and samples

$$\pi_t = \text{GS}(\ell_t, \tau) \in \Delta^{K-1}, \tag{3}$$

where τ is the temperature. The forward pass uses a one-hot sample, while the backward pass uses the continuous relaxation so that gradients can flow through attack choice. All K primitives are evaluated in parallel with parameters $\lambda_t^{(i)} \sim \mathcal{U}(\Lambda_i)$, where $\mathcal{U}$ denotes the uniform distribution over the valid range Λ_i, and their outputs are combined as

$$x_{t+1} = \sum_{i=1}^{K} \pi_t^{(i)} \, f_i(x_t, \, \lambda_t^{(i)}). \tag{4}$$

Relaxed inner maximization. Combining equation 2 and equation 4, the original min-max problem becomes

$$\min_{\theta,\psi} \max_{\phi} \; \mathbb{E}_{x,m,\, a\sim\pi_\phi} \Big[\mathcal{L}_{\text{msg}}\big(D_\psi(x_T), \, m\big) \Big], \tag{5}$$

where x_T is the attacked image after T sequential steps and $a = (a_0, \ldots, a_{T-1})$ denotes the full attack sequence. Equivalently, for fixed watermark parameters (θ, ψ), the adversary solves

$$\max_{\phi} \; \mathbb{E}_{a\sim\pi_\phi} \big[\mathcal{L}_{\text{msg}}\big(D_\psi(x_T), \, m\big) \big], \tag{6}$$

This highlights the main novelty of our setting: the adversary must learn not only which corruption is harmful, but also which order of corruptions jointly maximize decoder failure. Because attack choice is relaxed and differentiable, the controller can be updated jointly with the watermark model rather than through a separate black-box search over attack sequences.

2.2 Entropy Regularization

Without additional constraints, the inner maximization in Eq. equation 6 can collapse to a single maximally damaging attack sequence,

$$a^\star = \arg\max_{a} \; \mathcal{L}_{\text{msg}}\big(D_\psi(x_T), \, m\big), \tag{7}$$

which causes the adversary to over-specialize and reduces exploration as training evolves. To prevent this, we add an entropy bonus to the attack policy and optimize

$$\max_{\phi} \; \mathbb{E}_{a\sim\pi_\phi} \big[\mathcal{L}_{\text{msg}}\big(D_\psi(x_T), \, m\big) \big] + \lambda_{\text{ent}} \, \mathcal{H}(\pi_\phi), \tag{8}$$

where $\mathcal{H}(\pi_\phi)$ is the Shannon entropy of the attack policy and λ_{ent} controls the exploration–strength tradeoff. In practice, this keeps the controller exploratory early in training, avoids premature collapse to a trivially destructive strategy, and improves robustness beyond a single dominant failure mode. Full implementation details, including the per-step entropy decomposition and the exact optimization objective, are provided in Appendix B.

3 Related Work

Deep Learning-Based Watermarking A more comprehensive review appears in Appendix A. Deep neural networks have become the dominant approach to image and video watermarking, largely following the encoder–noise–decoder paradigm introduced by HiDDeN (Zhu et al., 2018). In this framework, an encoder embeds a message into an image, a noise layer simulates distortions, and a decoder recovers the message, enabling end-to-end optimization of robustness and imperceptibility. This formulation has been widely applied to images in MBRS (Jia et al., 2021), TrustMark (Bui et al., 2025), and InvisMark (Xu et al., 2025), and to video through DVMark (Luo et al., 2023), VideoSeal (Fernandez et al., 2024), and PixelSeal (Souček et al., 2025). A further line of work extends the same paradigm to in-generation watermarking methods such as Stable Signature (Fernandez et al., 2023), Tree-Ring (Wen et al., 2023), and WMAR (Jovanović et al., 2025). Despite these architectural advances, robustness is still typically trained through fixed or randomly sampled augmentations. That strategy helps against common isolated corruptions, but it under-covers realistic attack pipelines in which operation identity, parameter range, and order all matter. Our work addresses this training gap by replacing blind random augmentation with a learned compositional adversary that adaptively targets current failure modes.

Adversarial Training for Watermarking. While adversarial training has been explored in watermarking, existing approaches remain limited in scope. Many works introduce adversarial components but target objectives orthogonal to post-processing robustness. For example, HiDDeN (Zhu et al., 2018) and PixelSeal (Souček et al., 2025) employ discriminators to enforce visual imperceptibility rather than robustness; ROBIN (Huang et al., 2024) focuses on watermark concealment during diffusion generation; and InvZW citeptanvir2025invzw applies adversarial learning to zero-watermarking, which avoids modifying the image and therefore bypasses the encoder–extractor robustness tradeoff entirely. Methods that explicitly target robustness, such as RoMark citepwen2019romark, formulate watermarking as a min–max optimization problem but are not end-to-end differentiable and require repeated decoder evaluations to approximate worst-case augmentations, which makes training expensive and still leaves sequential compositions largely unmodeled. Related methods developed for image-classification robustness, including GAT citephsiung2023towards and Adversarial AutoAugment citepzhang2019adversarial, also rely on iterative inner optimization or multiple policy-augmented evaluations. These methods are useful algorithmic references, but their evaluation costs do not fit the long, high-resolution training workloads of watermarking; Appendix D.4 quantifies this difference. CAT differs from these approaches by providing adaptive adversarial training over compositional attacks within a shared training pass, making the approach practical at watermark scale.

4 Experiments

4.1 Architecture and Implementation Details

Architecture and training details. We evaluate CAT on VideoSeal 0.0, VideoSeal 1.0 (Fernandez et al., 2024), and PixelSeal (Souček et al., 2025). The underlying embedder and extractor follow the original PixelSeal-style architectures, while the learned adversary uses frozen DINOv3 ViT-S/16 features, a lightweight GRU for attack selection, and a lightweight MLP attack head to select augmentations sequentially. In total, the adversary contributes only 800K trainable parameters and modest memory overhead. Full architecture and training hyperparameters are deferred to Appendix B.

At each adversarial step t, visual features $z_t \in \mathbb{R}^{d_h}$ are extracted from x_t using a frozen DINOv3 ViT-S/16 backbone (Siméoni et al., 2025) followed by a two-layer MLP projection, and a GRU updates

$$h_t = \mathrm{GRU}(z_t, h_{t-1}), \quad h_0 = \mathbf{0}, \tag{9}$$

so that attack choices can depend on previously applied transformations. This recurrent controller is the core architectural ingredient needed to study sequential, order-sensitive attacks; implementation specifics are given in Appendix B.

Evaluation protocol. For each learned backbone, we compare CAT against a baseline trained for the same number of gradient updates and with the same augmentation budget, but without the learned adversary. We also compare to the pretrained variants of MBRS (Jia et al., 2021), TrustMark (Bui et al., 2025), and InvisMark (Xu et al., 2025). The single-step attack setting corresponds to $T = 1$, while the compositional attack setting corresponds to $T = 2$ and therefore a two-step attack sequence. Unless otherwise noted, the tables report bit accuracy and watermark capacity for each attack family, along with their overall averages; higher is better for both metrics. In the main image experiments, we evaluate on 1000 in-distribution (ID) SA-1B test images (Kirillov et al., 2023) and on the out-of-distribution (OOD) CLIC benchmark (Aytekin et al., 2019). This same-update, same-augmentation-budget comparison ensures that any gains from CAT are not simply due to additional augmentation budget, but to adaptive adversarial training itself. Appendix C analyzes the policy structure, Appendix D compares feasible adaptive alternatives, and Appendix D.4 analyzes RoMark-style search, GAT, and Adversarial AutoAugment. We report the latter as cost analyses rather than full performance baselines because their repeated decoder evaluations, inner optimization, or policy copies do not fit the watermark-scale training regime used here.

4.2 Robustness to Single-Step Attacks

CAT is most effective on challenging attack settings. Table 1 shows that even a single learned attack step improves robustness under the same-update, same-augmentation-budget comparison. On

SA-1B, CAT improves overall capacity for all three learned backbones, from 84.90 to 87.28 for VideoSeal 0.0, from 114.78 to 138.98 for VideoSeal 1.0, and from 61.52 to 105.20 for PixelSeal. The same pattern holds on CLIC, where overall capacity rises from 77.16 to 80.29 for VideoSeal 0.0, from 103.10 to 118.1 for VideoSeal 1.0, and from 54.67 to 95.28 for PixelSeal. The gains are concentrated in the more difficult corruption families, especially geometric and combined attacks, where random augmentation is least likely to sample the failure-inducing cases that matter most.

The primary empirical benefit is increased usable capacity. CAT yields not only improvements in bit accuracy, but also substantial gains in usable watermark capacity. This effect is especially pronounced for PixelSeal, whose overall capacity rises by 43.68 points on SA-1B and by 40.61 points on CLIC, and for VideoSeal 1.0, whose overall capacity improves by 24.20 points on SA-1B and 15.00 points on CLIC. Here, capacity measures the effective number of reliably recoverable bits under attack, not just whether decoding succeeds on average. This distinction matters because capacity increases sharply as bit accuracy approaches 100%, so even modest average accuracy gains can correspond to much larger capacity gains once performance enters the high-accuracy regime. The table also shows that CAT-trained models remain competitive with pretrained baselines on the main image benchmark: PixelSeal with CAT reaches 105.20 overall capacity on SA-1B, compared with 70.48 for InvisMark and 75.99 for TrustMark.

Table 1: **A single learned attack step already improves robustness over random augmentation training, with the largest gains for VideoSeal 1.0 and PixelSeal, especially on the most difficult combined attacks and under severe distribution shift.** Single-step attack accuracy and watermark capacity on in-distribution SA-1B and out-of-distribution CLIC images. Each cell reports bit accuracy and capacity for identity, value, compression, and geometric augmentation families, plus the combined setting. Superscripts report the standard deviation over three seeds.

		Identity		Value		Compression		Geometric		Combined		Overall	
	Model (bits)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)
SA-1B (ID)	InvisMark (100)	99.0	95.99	87.6	72.90	95.4	88.45	82.8	63.25	86.1	67.84	86.6	70.48
	TrustMark (100)	99.6	98.92	95.6	86.28	89.8	79.09	75.4	50.29	99.3	97.90	88.9	75.99
	MBRS (256)	98.7	242.60	91.5	185.99	88.4	190.52	65.3	78.90	95.9	217.48	83.3	157.97
	VideoSeal 0.0 (96)	99.7	94.00	98.4	88.69	98.0	86.86	94.5	75.58	99.4	92.33	97.3	84.90
	+ FAA	99.4	92.78	96.6	83.67	82.8	60.89	61.1	21.33	99.0	90.81	74.0	44.31
	+ CAT	$99.8^{(0.0)}$	$94.9^{(0.1)}$	$98.3^{(0.5)}$	$88.8^{(1.7)}$	$98.9^{(0.2)}$	$90.3^{(1.0)}$	$95.8^{(0.5)}$	$80.5^{(2.0)}$	$99.5^{(0.2)}$	$93.3^{(0.9)}$	**97.8**$^{(0.4)}$ +0.6%	**87.28**$^{(1.7)}$ +2.8%
	VideoSeal 1.0 (256)	89.8	135.17	87.9	123.45	87.2	118.96	82.2	94.39	89.2	130.60	86.2	114.78
	+ FAA	99.5	248.65	95.4	210.59	82.3	157.47	60.8	53.59	97.6	226.46	73.4	112.19
	+ CAT	$93.4^{(0.7)}$	$168.1^{(7.1)}$	$89.5^{(1.7)}$	$143.6^{(5.9)}$	$90.4^{(0.5)}$	$146.8^{(4.3)}$	$85.5^{(1.1)}$	$120.6^{(3.3)}$	$91.7^{(0.9)}$	$154.5^{(8.2)}$	**88.8**$^{(1.1)}$ +3.0%	**138.98**$^{(4.2)}$ +21.1%
	PixelSeal (128)	91.8	76.46	89.5	68.65	88.0	63.89	81.9	48.19	90.2	70.27	86.9	61.52
	+ FAA	93.5	87.32	87.3	66.91	77.4	51.87	59.2	17.88	89.1	70.02	69.6	36.40
	+ CAT	$99.3^{(0.6)}$	$122.7^{(4.3)}$	$96.9^{(1.1)}$	$109.9^{(6.2)}$	$97.1^{(1.2)}$	$110.3^{(6.4)}$	$92.1^{(1.8)}$	$91.4^{(7.4)}$	$98.3^{(1.0)}$	$116.0^{(6.3)}$	**95.6**$^{(1.0)}$ +10.1%	**105.20**$^{(6.8)}$ +71.0%
CLIC (OOD)	InvisMark (100)	99.0	96.02	87.6	72.93	69.1	35.10	82.5	62.83	58.2	19.23	76.6	52.09
	TrustMark (100)	99.3	98.39	95.9	87.60	89.4	78.18	75.3	50.21	94.7	84.07	87.8	72.87
	MBRS (256)	98.7	243.37	92.2	191.47	84.8	162.21	64.8	74.77	77.5	105.06	78.4	127.42
	VideoSeal 0.0 (96)	99.8	94.59	98.6	89.85	97.7	86.11	94.8	76.61	84.4	57.95	93.8	77.16
	+ FAA	99.4	92.68	93.0	73.55	77.7	47.50	55.3	7.12	90.8	66.53	68.6	30.67
	+ CAT	$99.9^{(0.0)}$	$95.4^{(0.2)}$	$98.3^{(0.5)}$	$88.0^{(2.2)}$	$99.0^{(0.4)}$	$92.2^{(1.5)}$	$95.5^{(0.4)}$	$80.3^{(0.8)}$	$88.0^{(1.5)}$	$64.7^{(3.3)}$	**94.9**$^{(0.4)}$ +1.3%	**80.29**$^{(1.8)}$ +4.1%
	VideoSeal 1.0 (256)	90.0	136.47	88.3	125.61	86.9	117.38	82.4	95.59	76.6	78.96	83.3	103.10
	+ FAA	99.3	246.75	89.3	170.49	76.4	116.75	55.5	18.62	84.4	135.91	67.5	72.79
	+ CAT	$93.9^{(0.8)}$	$172.3^{(8.6)}$	$88.4^{(2.8)}$	$141.9^{(8.6)}$	$92.7^{(1.5)}$	$164.5^{(7.7)}$	$80.0^{(0.5)}$	$108.8^{(3.9)}$	$82.0^{(1.9)}$	$78.2^{(3.5)}$	**84.5**$^{(0.6)}$ +1.5%	**118.1**$^{(3.5)}$ +14.6%
	PixelSeal (128)	92.1	77.60	90.0	70.26	87.1	61.44	82.2	49.03	74.8	40.15	83.3	54.67
	+ FAA	93.5	87.36	80.5	51.64	71.3	36.67	54.6	6.94	73.2	32.37	63.8	22.67
	+ CAT	$99.6^{(0.4)}$	$124.7^{(2.9)}$	$96.4^{(0.8)}$	$109.9^{(4.5)}$	$96.3^{(2.0)}$	$111.9^{(7.8)}$	$86.6^{(0.4)}$	$86.3^{(1.8)}$	$86.5^{(0.5)}$	$81.4^{(1.0)}$	**90.7**$^{(0.4)}$ +4.8%	**95.28**$^{(1.5)}$ +74.3%

4.3 Robustness to Compositional Attacks

CAT is particularly beneficial for compositional attacks. The advantage of CAT becomes clearer when the adversary applies a two-step sequence ($T = 2$) and must model attack identity and order. Table 2 shows overall capacity gains for every learned backbone on both datasets: on SA-1B, capacity rises from 80.50 to 86.56 for VideoSeal 0.0, from 105.25 to 130.74 for VideoSeal 1.0, and from 103.13 to 109.57 for PixelSeal; on CLIC, it rises from 79.41 to 87.66, from 106.17 to 120.62, and from 99.95 to 106.68, respectively. The gains concentrate on harder mixed and repeated pairs: on SA-1B, VideoSeal 1.0 improves from 134.39 to 166.4 on Comp+Comp and from 109.22 to 138.4 on Comp+Geom, while PixelSeal improves from 117.75 to 122.8 and from 106.43 to 112.9 on those pairs. CAT-trained PixelSeal reaches 109.57 overall capacity on SA-1B and 106.68 on CLIC, exceeding both pretrained baselines and showing the benefit of adapting to order-sensitive compositions.

A fixed proxy policy misses evolving geometric failures. Faster AutoAugment (FAA) (Hataya et al., 2020) searches once with a 32-bit proxy and then freezes its policy for watermark training. Because the policy is learned once on a proxy task and then frozen during watermark training, FAA cannot adapt to newly emerging watermark-model failures and is weak on the hardest geometric

Table 2: **Under compositional attacks, CAT yields improvements in usable watermark capacity, especially for VideoSeal 1.0 and PixelSeal on hard mixed and repeated attack pairs.** Compositional-attack results on in-distribution SA-1B and out-of-distribution CLIC images for two-step attack sequences. Each cell reports bit accuracy and capacity for pairwise augmentation-family compositions. Superscripts report the standard deviation over three seeds.

		Val+Val		Val+Comp		Val+Geom		Comp+Comp		Comp+Geom		Geom+Geom		Overall	
	Model (bits)	Bit acc. (%) ↑	Capacity ↑	Bit acc. (%) ↑	Capacity ↑	Bit acc. (%) ↑	Capacity ↑	Bit acc. (%) ↑	Capacity ↑	Bit acc. (%) ↑	Capacity ↑	Bit acc. (%) ↑	Capacity ↑	Bit acc. (%) ↑	Capacity ↑
SA-1B (ID)	InvisMark (100)	89.8	71.44	65.2	27.15	87.5	68.63	47.4	0.85	60.3	21.75	88.8	70.52	78.6	52.60
	TrustMark (100)	95.7	86.21	96.5	88.49	78.6	52.86	97.4	90.92	77.9	50.97	70.8	36.58	82.3	59.88
	MBRS (256)	91.7	182.87	83.6	130.67	60.2	41.86	77.4	90.86	55.4	15.81	49.5	0.98	65.0	60.56
	VideoSeal 0.0 (96)	97.2	83.44	98.5	87.84	96.1	78.08	99.2	91.40	97.4	82.79	93.0	72.83	96.4	80.50
	+ FAA	99.8	95.14	97.5	88.16	83.4	63.74	61.5	22.80	99.7	94.41	74.5	46.74	88.6	73.12
	+ CAT	$98.3^{(0.6)}$	$88.9^{(2.1)}$	$99.4^{(0.2)}$	$92.8^{(1.1)}$	$97.5^{(1.0)}$	$85.2^{(3.6)}$	$99.8^{(0.1)}$	$95.0^{(0.6)}$	$98.7^{(0.8)}$	$89.7^{(3.1)}$	$93.3^{(1.4)}$	$77.2^{(3.9)}$	**97.5**$^{(1.0)}$ +1.1%	**86.56**$^{(3.2)}$ +7.3%
	VideoSeal 1.0 (256)	87.5	121.35	88.7	127.69	82.7	95.41	89.7	134.39	85.8	109.22	79.9	86.78	84.4	105.25
	+ FAA	68.7	30.72	67.8	29.11	62.3	22.27	53.8	11.32	68.5	30.39	58.9	17.78	64.3	24.83
	+ CAT	$88.5^{(2.2)}$	$138.6^{(7.6)}$	$91.3^{(0.8)}$	$153.6^{(4.6)}$	$86.0^{(1.9)}$	$121.2^{(5.7)}$	$93.2^{(0.7)}$	$166.4^{(7.1)}$	$89.3^{(0.4)}$	$138.4^{(4.2)}$	$83.0^{(0.5)}$	$111.5^{(2.5)}$	**87.4**$^{(1.7)}$ +3.5%	**130.74**$^{(5.7)}$ +24.2%
	PixelSeal (128)	96.4	106.45	97.7	112.06	95.4	100.25	98.7	117.75	96.8	106.43	92.2	93.04	95.7	103.13
	+ FAA	86.2	55.85	83.1	48.89	73.5	36.19	57.5	12.69	85.7	54.44	66.9	26.20	77.3	41.62
	+ CAT	$97.1^{(0.7)}$	$111.2^{(4.1)}$	$98.5^{(0.6)}$	$117.7^{(4.2)}$	$96.4^{(0.9)}$	$107.3^{(5.6)}$	$99.4^{(0.4)}$	$122.8^{(2.9)}$	$96.9^{(1.7)}$	$112.9^{(5.4)}$	$92.7^{(0.9)}$	$100.1^{(1.3)}$	**96.4**$^{(0.9)}$ +0.7%	**109.57**$^{(4.2)}$ +6.2%
CLIC (OOD)	InvisMark (100)	90.2	72.69	65.3	27.61	87.5	68.97	47.4	0.85	60.0	21.29	88.3	69.50	78.5	52.68
	TrustMark (100)	96.4	88.76	96.8	89.85	79.0	53.95	97.4	91.65	78.2	51.93	71.0	37.13	82.6	61.05
	MBRS (256)	92.7	190.84	85.1	140.04	60.8	45.61	79.1	100.83	55.9	18.03	49.6	0.97	65.7	64.90
	VideoSeal 0.0 (96)	97.8	86.28	96.1	81.15	96.8	80.81	93.7	77.05	94.7	75.52	93.7	74.67	95.9	79.41
	+ FAA	99.8	94.90	92.9	76.36	78.4	50.21	55.9	8.27	92.3	73.82	69.2	32.71	83.6	59.86
	+ CAT	$99.1^{(0.2)}$	$91.5^{(1.0)}$	$98.7^{(0.4)}$	$90.4^{(1.4)}$	$97.9^{(0.6)}$	$86.6^{(2.4)}$	$98.9^{(0.5)}$	$91.2^{(1.8)}$	$98.2^{(0.8)}$	$87.9^{(2.8)}$	$97.0^{(1.4)}$	$84.1^{(4.0)}$	**98.1**$^{(0.7)}$ +2.3%	**87.66**$^{(2.4)}$ +10.4%
	VideoSeal 1.0 (256)	88.1	124.44	88.6	127.38	83.2	98.03	89.5	133.09	85.4	106.99	80.0	87.42	84.6	106.17
	+ FAA	68.3	29.78	65.9	25.55	60.1	18.07	51.7	7.07	66.0	25.43	56.7	13.65	62.3	20.94
	+ CAT	$85.1^{(0.1)}$	$123.9^{(3.0)}$	$86.3^{(0.9)}$	$131.0^{(7.4)}$	$83.2^{(1.0)}$	$113.3^{(5.4)}$	$89.6^{(1.4)}$	$147.5^{(10.1)}$	$87.3^{(0.7)}$	$130.1^{(6.4)}$	$83.7^{(1.0)}$	$110.7^{(8.7)}$	**84.9**$^{(0.8)}$ +0.4%	**120.62**$^{(5.8)}$ +13.6%
	PixelSeal (128)	97.3	111.00	93.9	98.69	96.1	103.61	90.2	90.40	92.6	92.72	92.7	94.93	94.6	99.95
	+ FAA	86.0	55.23	79.3	41.57	69.3	28.20	53.3	5.31	79.6	41.46	62.7	18.48	73.2	33.65
	+ CAT	$98.2^{(0.2)}$	$115.6^{(1.7)}$	$97.0^{(0.8)}$	$109.4^{(1.3)}$	$95.3^{(1.2)}$	$102.8^{(5.5)}$	$96.5^{(1.5)}$	$106.2^{(1.8)}$	$96.4^{(0.6)}$	$108.1^{(1.9)}$	$96.5^{(0.1)}$	$105.5^{(2.5)}$	**96.3**$^{(0.7)}$ +1.8%	**106.68**$^{(3.1)}$ +6.7%

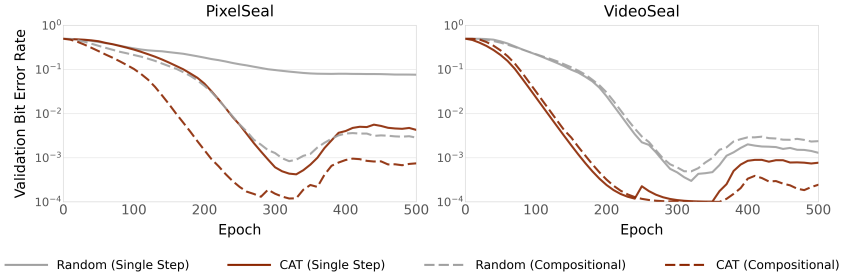


Figure 4: **CAT substantially accelerates convergence for both PixelSeal and VideoSeal, and this advantage persists from single-step to compositional training.** Validation bit error over training for PixelSeal (left) and VideoSeal (right). Solid curves show single-step training and dashed curves show compositional training; gray curves are random augmentation and brown curves are CAT. Across both backbones, CAT reaches lower validation bit error earlier, indicating that CAT identifies and improves upon informative training-time failures more effectively than random augmentation.

cases: on SA-1B single-step attacks, its geometric accuracy/capacity are 61.1/21.33, 60.8/53.59, and 59.2/17.88 for VideoSeal 0.0, VideoSeal 1.0, and PixelSeal, respectively, whereas CAT reaches 95.8/80.5, 85.5/120.6, and 92.1/91.4. CAT updates the policy jointly with the watermark model and conditions each sequential decision on the current intermediate image, allowing it to follow newly exposed failure modes and target difficult geometric and mixed attacks.

4.4 Validation Convergence

CAT improves sample efficiency by focusing training on informative failures. Figure 4 shows that CAT consistently reaches lower validation bit error earlier than random augmentation for both PixelSeal and VideoSeal, and that this advantage persists from the single-step setting to the compositional setting. The gap is especially meaningful because it appears throughout training rather than only at the end: the learned adversary exposes failure cases that are already relevant to the model's current weaknesses, whereas random augmentation spends many updates on transformations that are either too easy or insufficiently targeted. This makes optimization more sample efficient, not merely more aggressive. The result is particularly important in the compositional attack regime, where random augmentation must discover useful short attack compositions by chance, while CAT can adaptively concentrate on the sequences that actually break decoding. Because our method reuses the standard training pass instead of introducing separate inner attack search, the lower validation bit error reflects improved optimization-step efficiency: CAT allocates robust supervision more effectively at the same number of training updates.

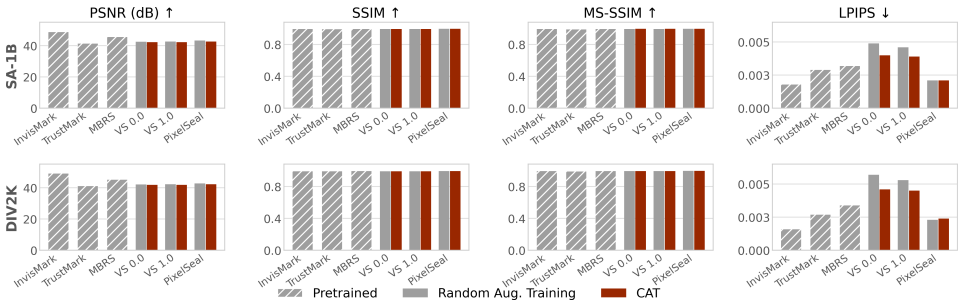


Figure 5: **CAT improves robustness with little to no change in image fidelity.** PSNR, SSIM, MS-SSIM, and LPIPS on SA-1B (top) and DIV2K (bottom). Gray: random augmentation; orange: CAT; hatched: pretrained baselines. VS denotes VideoSeal; arrows indicate preferred directions.

4.5 Image Quality

Figure 5 compares image fidelity on SA-1B and DIV2K. CAT preserves similar PSNR, SSIM, and MS-SSIM to random augmentation while improving robustness. LPIPS improves for both VideoSeal variants; for PixelSeal, it is unchanged on SA-1B and increases by 0.0001 on DIV2K. These results suggest that the robustness gains do not require substantially more visible watermarks.

4.6 Ablations

Ablation results. Sequential factorization and state conditioning are critical to CAT: predicting the complete attack composition as a single action performs worse and collapses policy entropy, while removing intermediate-image conditioning reduces robustness on difficult compositional attacks. UCB, curriculum-based sampling, and RL also underperform CAT, indicating that neither harder attack scheduling nor generic adaptive selection alone explains CAT's gains. We also ablate entropy regularization, the frozen DINOv3 backbone, and differentiable Gumbel-Softmax selection. Tables 3 and 4 show that removing entropy causes the largest robustness collapse. Replacing DINOv3 with ResNet18 lowers single-step overall bit accuracy/capacity from 95.6/105.20 to 88.4/80.66, while compositional bit accuracy is nearly unchanged (96.4 versus 96.5) and capacity decreases slightly.

Table 3: Ablation study on SA-1B for single-step attacks. Each cell reports bit accuracy and watermark capacity for the indicated augmentation family.

Method	Identity		Value		Compression		Geometric		Combined		Overall	
	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)
CAT	99.3	122.7	96.9	109.9	97.1	110.3	92.1	91.4	98.3	116.0	**95.6**	**105.20**
- Entropy Reg.	98.7	116.32	94.0	95.50	98.5	115.24	67.8	36.62	97.7	110.03	86.5	81.40
+ ResNet18	96.9	106.40	92.3	88.22	96.1	102.47	77.6	55.04	95.0	96.16	88.4	80.66
+ UCB	99.7	125.09	94.9	103.02	99.4	123.63	80.8	70.69	97.9	114.24	91.4	97.36

Table 4: Ablation study on SA-1B for compositional attacks. Each cell reports bit accuracy and watermark capacity for the indicated family pair.

Method	Val+Val		Val+Comp		Val+Geom		Comp+Comp		Comp+Geom		Geom+Geom		Overall	
	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)
CAT	97.1	111.2	98.5	117.7	96.4	107.3	99.4	122.8	96.9	112.9	92.7	100.1	96.4	**109.57**
- Entropy Reg.	50.4	0.71	50.5	0.71	50.3	0.72	50.5	0.70	50.3	0.72	50.3	0.73	50.3	0.72
+ ResNet18	97.4	112.73	98.6	118.54	96.3	106.29	99.5	124.10	97.9	113.81	92.3	97.12	**96.5**	109.21
+ UCB	96.5	106.95	97.8	113.30	87.6	79.10	98.8	119.05	89.3	85.13	81.7	65.51	89.8	87.24

5 Conclusion

CAT is a sequential differentiable adversarial training framework that learns which realistic transformations to apply, and in what order, to expose watermarking failure cases that random augmentation under-samples. Across image, video, and autoregressive watermarking, CAT improves robustness and usable capacity, converges faster, and preserves inference-time behavior and perceptual quality, with the strongest gains on geometric, compositional, and out-of-distribution attacks. These results show that robust visual watermarking benefits from adaptive compositional adversaries that respond to evolving model weaknesses and the realized effects of preceding attacks, rather than static and/or independent random corruptions.

Reproducibility Statement

The paper and appendices specify the watermarking models and payloads, attack transformations and compositions, training settings, datasets, and evaluation protocols used in the experiments. Results from repeated runs identify the number of seeds in the relevant tables; the main image results report standard deviations over three seeds where applicable.

AI Use Disclosure

Generative AI assisted with language editing, code writing, manuscript organization, title development, and presentation of existing results. It did not generate data, derive results, or create citations. The authors verified all revisions and take responsibility for the text, claims, and artifacts.

References

Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp. 126–135, 2017.

Mahdi Ahmadi, Alireza Norouzi, Nader Karimi, Shadrokh Samavi, and Ali Emami. Redmark: Framework for residual diffusion watermarking based on deep networks. Expert Systems with Applications, 146:113157, 2020.

Bang An, Mucong Ding, Tahseen Rabbani, Aakriti Agrawal, Yuancheng Xu, Chenghao Deng, Sicheng Zhu, Abdirisak Mohamed, Yuxin Wen, Tom Goldstein, et al. Waves: Benchmarking the robustness of image watermarks. In International Conference on Machine Learning, pp. 1456–1492. PMLR, 2024.

Caglar Aytekin, Francesco Cricri, Antti Hallapuro, Jani Lainema, Emre Aksu, and Miska M Hannuksela. A compression objective and a cycle loss for neural image compression. In CVPR Workshops, pp. 0, 2019.

Mauro Barni, Franco Bartolini, and Alessandro Piva. Improved wavelet-based watermarking through pixel-wise masking. IEEE transactions on image processing, 10(5):783–791, 2001.

Adrian G Bors and Ioannis Pitas. Image watermarking using dct domain constraints. In Proceedings of 3rd IEEE International Conference on Image Processing, volume 3, pp. 231–234. IEEE, 1996.

Tu Bui, Shruti Agarwal, and John Collomosse. Trustmark: Robust watermarking and watermark removal for arbitrary resolution images. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 18629–18639, 2025.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In 2017 ieee symposium on security and privacy (sp), pp. 39–57. Ieee, 2017.

Luan Chen, Chengyou Wang, Xiao Zhou, and Zhiliang Qin. Robust and compatible video watermarking via spatio-temporal enhancement and multiscale pyramid attention. IEEE Transactions on Circuits and Systems for Video Technology, 35(2):1548–1561, 2024.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In International conference on machine learning, pp. 2206–2216. PMLR, 2020.

Francesco Croce and Matthias Hein. Adversarial robustness against multiple and single l_p-threat models via quick fine-tuning of robust classifiers. In International Conference on Machine Learning, pp. 4436–4454. PMLR, 2022.

Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 113–123, 2019.

Mucong Ding, Tahseen Rabbani, Bang An, Souradip Chakraborty, Chenghao Deng, Mehrdad Saberi, Yuxin Wen, Xuandong Zhao, Mo Zhou, Anirudh Satheesh, et al. Erasing the invisible: A stress-test challenge for image watermarks. In NeurIPS 2024 Competition Track, 2024.

Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 12873–12883, 2021.

Pierre Fernandez, Guillaume Couairon, Hervé Jégou, Matthijs Douze, and Teddy Furon. The stable signature: Rooting watermarks in latent diffusion models. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 22466–22477, 2023.

Pierre Fernandez, Hady Elsahar, I Zeki Yalniz, and Alexandre Mourachko. Video seal: Open and efficient video watermarking. arXiv preprint arXiv:2412.09492, 2024.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572, 2014.

Ryuichiro Hataya, Jan Zdenek, Kazuki Yoshizoe, and Hideki Nakayama. Faster autoaugment: Learning augmentation strategies using backpropagation. In European Conference on Computer Vision, 2020.

Daniel Ho, Eric Liang, Xi Chen, Ion Stoica, and Pieter Abbeel. Population based augmentation: Efficient learning of augmentation policy schedules. In International conference on machine learning, pp. 2731–2741. PMLR, 2019.

Lei Hsiung, Yun-Yun Tsai, Pin-Yu Chen, and Tsung-Yi Ho. Towards compositional adversarial robustness: Generalizing adversarial training to composite semantic perturbations. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 24658–24667, 2023.

Kun Hu, Zixuan Hu, Qianhui Zhu, Xiaochao Wang, and Xingjun Wang. Stegavideo: Robust high-resolution video steganography with temporal and edge guidance. In Pacific Conference on Computer Graphics and Applications (Huangshan, CN)(PG'24). EG, Eindhoven, NL, Article, volume 1280, pp. 12, 2024.

Huayang Huang, Yu Wu, and Qian Wang. Robin: Robust and invisible watermarks for diffusion models with adversarial optimization. Advances in Neural Information Processing Systems, 37: 3937–3963, 2024.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In International Conference on Learning Representations, 2017.

Zhaoyang Jia, Han Fang, and Weiming Zhang. Mbrs: Enhancing robustness of dnn-based watermarking by mini-batch of real and simulated jpeg compression. In Proceedings of the 29th ACM international conference on multimedia, pp. 41–49, 2021.

Nikola Jovanović, Ismail Labiad, Tomáš Souček, Martin Vechev, and Pierre Fernandez. Watermarking autoregressive image generation. arXiv preprint arXiv:2506.16349, 2025.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 4015–4026, 2023.

Cassidy Laidlaw, Sahil Singla, and Soheil Feizi. Perceptual adversarial robustness: Defense against unseen threat models. arXiv preprint arXiv:2006.12655, 2020.

Yonggang Li, Guosheng Hu, Yongtao Wang, Timothy Hospedales, Neil M Robertson, and Yongxin Yang. Differentiable automatic data augmentation. In European conference on computer vision, pp. 580–595. Springer, 2020.

Sungbin Lim, Ildoo Kim, Taesup Kim, Chiheon Kim, and Sungwoong Kim. Fast autoaugment. In Advances in Neural Information Processing Systems, 2019.

Aoming Liu, Zehao Huang, Zhiwu Huang, and Naiyan Wang. Direct differentiable augmentation search. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 12219–12228, 2021.

Shilin Lu, Zihan Zhou, Jiayou Lu, Yuanzhi Zhu, and Adams Wai-Kin Kong. Robust watermarking using generative priors against image editing: From benchmarking to advances. arXiv preprint arXiv:2410.18775, 2024.

Xiyang Luo, Yinxiao Li, Huiwen Chang, Ce Liu, Peyman Milanfar, and Feng Yang. Dvmark: a deep multiscale framework for video watermarking. IEEE Transactions on Image Processing, 2023.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083, 2017.

Pratyush Maini, Eric Wong, and Zico Kolter. Adversarial robustness against the union of multiple perturbation models. In International Conference on Machine Learning, pp. 6640–6650. PMLR, 2020.

Aayush Mishra, Suraj Kumar, Aditya Nigam, and Saiful Islam. Vstegnet: Video steganography network using spatio-temporal features and micro-bottleneck. In BMVC, volume 274, 2019.

Samuel G Müller and Frank Hutter. Trivialaugment: Tuning-free yet state-of-the-art data augmentation. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 774–782, 2021.

Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. arXiv preprint arXiv:2205.07460, 2022.

Alessandro Piva, Mauro Barni, Franco Bartolini, and Vito Cappellini. Dct-based watermark recovering without resorting to the uncorrupted original image. In Proceedings of international conference on image processing, volume 1, pp. 520–523. IEEE, 1997.

Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. arXiv preprint arXiv:2410.13720, 2024.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. arXiv preprint arXiv:2408.00714, 2024.

Luke Rowe, Benjamin Thérien, Krzysztof Czarnecki, and Hongyang Zhang. A closer look at robustness to l-infinity and spatial perturbations and their composition. arXiv preprint arXiv:2210.02577, 2022.

Mehrdad Saberi, Vinu Sankar Sadasivan, Keivan Rezaei, Aounon Kumar, Atoosa Chegini, Wenxiao Wang, and Soheil Feizi. Robustness of ai-image detectors: Fundamental limits and practical attacks. arXiv preprint arXiv:2310.00076, 2023.

Xiaofeng Shen, Heng Yao, Shunquan Tan, and Chuan Qin. Vhnet: A video hiding network with robustness to video coding. Journal of Information Security and Applications, 75:103515, 2023.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. arXiv preprint arXiv:2508.10104, 2025.

Tomáš Souček, Pierre Fernandez, Hady Elsahar, Sylvestre-Alvise Rebuffi, Valeriu Lacatusu, Tuan Tran, Tom Sander, and Alexandre Mourachko. Pixel seal: Adversarial-only training for invisible image and video watermarking. arXiv preprint arXiv:2512.16874, 2025.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199, 2013.

Florian Tramer and Dan Boneh. Adversarial training and robustness for multiple perturbations. Advances in neural information processing systems, 32, 2019.

Bingyang Wen and Sergul Aydore. Romark: A robust watermarking system using adversarial training. arXiv preprint arXiv:1910.01221, 2019.

Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. Tree-ring watermarks: Fingerprints for diffusion images that are invisible and robust. arXiv preprint arXiv:2305.20030, 2023.

Xiang-Gen Xia, Charles G Boncelet, and Gonzalo R Arce. Wavelet transform based watermark for digital images. Optics Express, 3(12):497–511, 1998.

Rui Xu, Mengya Hu, Deren Lei, Yaxi Li, David Lowe, Alex Gorevski, Mingyu Wang, Emily Ching, and Alex Deng. Invismark: Invisible and robust watermarking for ai-generated image provenance. In 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 909–918. IEEE, 2025.

Guanhui Ye, Jiashi Gao, Yuchen Wang, Liyan Song, and Xuetao Wei. Itov: efficiently adapting deep learning-based image watermarking to video watermarking. In 2023 International Conference on Culture-Oriented Science and Technology (CoST), pp. 192–197. IEEE, 2023.

Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Randomized autoregressive visual generation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 18431–18441, 2025.

Honglei Zhang and Yidong Li. Digital watermarking via inverse gradient attention. In 2022 9th International Conference on Behavioural and Social Computing (BESC), pp. 1–3. IEEE, 2022.

Kevin Alex Zhang, Lei Xu, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Robust invisible video watermarking with attention. arXiv preprint arXiv:1909.01285, 2019a.

Xinyu Zhang, Qiang Wang, Jian Zhang, and Zhao Zhong. Adversarial autoaugment. arXiv preprint arXiv:1912.11188, 2019b.

Xuanyu Zhang, Runyi Li, Jiwen Yu, Youmin Xu, Weiqi Li, and Jian Zhang. Editguard: Versatile image watermarking for tamper localization and copyright protection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11964–11974, 2024.

Xuandong Zhao, Kexun Zhang, Zihao Su, Saastha Vasan, Ilya Grishchenko, Christopher Kruegel, Giovanni Vigna, Yu-Xiang Wang, and Lei Li. Invisible image watermarks are provably removable using generative ai. Advances in neural information processing systems, 37:8643–8672, 2024.

Jiren Zhu, Russell Kaplan, Justin Johnson, and Li Fei-Fei. Hidden: Hiding data with deep networks. In Proceedings of the European conference on computer vision (ECCV), pp. 657–672, 2018.

Appendix Contents

Appendix A Extended Related Work 15
Appendix B Implementation and Evaluation Details 16
B.1 Model Architectures and Training Setup 16
B.2 Entropy Regularization and Optimization Details 16
B.3 Evaluation Methodology 17
Appendix C Dissecting CAT: Sequential Factorization, State Conditioning, and Policy Stability 18
C.1 Sequential versus joint composition prediction 18
C.2 Intermediate-state conditioning 19
C.3 Policy stability and entropy dynamics 19
C.4 Visual encoder ablation 20
Appendix D Comparison with Alternative Adaptive and Adversarial Training Strategies 20
D.1 Handcrafted Difficulty Curriculum 20
D.2 UCB Attack-Frequency Baseline 20
D.3 Reinforcement-learning controller 21
D.4 Most existing adversarial augmentation methods don't scale 21
Appendix E Generalization to Longer Attack Chains 22
E.1 Generalization beyond the training depth 22
E.2 Simulated social-media pipelines 23
E.3 Native three-step training 25
Appendix F Generalization to Realistic Removal and Editing Attacks 25
Appendix G Generalization Across Watermarking Paradigms 26
G.1 Autoregressive watermarking setup 26
G.2 Taming and RAR-XL robustness 26
G.3 Synchronization and token alignment 26
Appendix H Additional OOD Image Results 29
Appendix I Additional Video Results 29
Appendix J Payload Scaling 31
Appendix K Forward and Backward Transfer 31
Appendix L Qualitative Results and Watermark Residual Analysis 33
L.1 Spatial and Frequency Analysis of Watermark Residuals 36
Appendix M Broader Impact 39

A Extended Related Work

Traditional Watermarking. Early watermarking methods embedded signals in spatial or frequency domains such as DCT (Bors & Pitas, 1996; Piva et al., 1997) and DWT (Barni et al., 2001; Xia et al., 1998), relying on hand-crafted rules designed for specific distortion types. While effective under controlled conditions, these approaches generally lack adaptability to diverse transformations. This motivated the development of model-based methods that incorporate differentiable transformations directly into training.

Model-based Image Watermarking. Deep neural networks have become the dominant approach to watermarking, largely following the encoder–noise–decoder framework introduced by HiDDeN (Zhu et al., 2018). In this formulation, the noise layer simulates distortions during training to improve robustness. Subsequent work extends this framework with architectural and training improvements, including residual diffusion embedding (Ahmadi et al., 2020), attention mechanisms (Zhang & Li, 2022), and improved robustness to compression via mini-batch mixing of real and simulated JPEG (Jia et al., 2021). More recent approaches address resolution-independent embedding (Bui et al., 2025; Xu et al., 2025), tamper localization (Zhang et al., 2024), and robustness to diffusion-based purification (Lu et al., 2024; Nie et al., 2022; Zhao et al., 2024; Saberi et al., 2023). In parallel, in-generation watermarking methods such as Stable Signature (Fernandez et al., 2023), Tree-Ring (Wen et al., 2023), and WMAR (Jovanović et al., 2025) embed watermarks directly during image synthesis. Many post-processing methods, including PixelSeal (Souček et al., 2025), TrustMark (Bui et al., 2025), InvisMark (Xu et al., 2025), and MBRS (Jia et al., 2021), are trained using noise layers composed of fixed or randomly sampled augmentations.

Model-based Video Watermarking. Video watermarking extends these ideas to the temporal domain but introduces additional computational challenges. Early approaches such as VStegNet (Mishra et al., 2019) and RivaGAN (Zhang et al., 2019a) adapted image-based architectures to video, often operating on full spatio-temporal tensors. Later works, including DVMark (Luo et al., 2023), VH-Net (Shen et al., 2023), RC-VWN (Chen et al., 2024), and StegaVideo (Hu et al., 2024), introduced improved spatio-temporal architectures, multiscale designs, and differentiable approximations of compression to enhance robustness and efficiency. Alternative approaches such as ItoV (Ye et al., 2023) and VideoSeal (Fernandez et al., 2024) reduce computational complexity by folding the temporal dimension into the channel dimension, enabling the use of image-based architectures for video watermarking. PixelSeal (Souček et al., 2025) also extends to video, using adversarial training with a discriminator to enforce perceptual quality.

Composite Adversaries. Adversarial training against a single perturbation type, most commonly an ℓ_p-norm ball, has been studied thoroughly (Goodfellow et al., 2014; Madry et al., 2017; Szegedy et al., 2013; Carlini & Wagner, 2017). However, robustness to a particular threat model does not generalize to overall robustness against all attacks. Several works have shown that training against one threat model can actually increase vulnerability to others (Tramer & Boneh, 2019; Maini et al., 2020). This has motivated a line of work on adversarial training against a union or composition of multiple perturbation types (Tramer & Boneh, 2019; Maini et al., 2020; Croce & Hein, 2020; 2022; Laidlaw et al., 2020; Rowe et al., 2022). Most closely related to our setting, Hsiung et al. (2023) proposed generalized adversarial training (GAT) over composite semantic perturbations, including combinations of hue, saturation, brightness, contrast, and rotation. In parallel, some literature has explored learning augmentation policies rather than fixing them by hand (Cubuk et al., 2019; Li et al., 2020; Liu et al., 2021; Ho et al., 2019; Müller & Hutter, 2021). Zhang et al. (2019b) further refined augmentation policy search into a min-max game between a target network and an augmentation network, jointly optimizing both. Rather than training for classification robustness or generalization accuracy, our work involves training an adversary designed to break a watermark encoder-extractor pair, operating over a structured space of post-processing operations relevant to the image and video distribution shift problem. Furthermore, unlike GAT (Hsiung et al., 2023), which schedules attack order as a discrete search problem, our adversary is a continuous, sequential, differentiable agent trained end-to-end via gradient flow, enabling it to discover attack compositions that are not enumerable by hand.

B Implementation and Evaluation Details

B.1 Model Architectures and Training Setup

Watermarking backbones and adversary architecture. The embedder and extractor use the same architectures as in (Souček et al., 2025): the embedder is a U-Net with 8 residual blocks and 16 base channels with multipliers $[1, 2, 4, 8]$, operating on the YUV luminance channel with group normalization and GELU activations, while the extractor uses a ConvNeXt-Tiny backbone with depths $[3, 3, 9, 3]$ and channel dimensions $[96, 192, 384, 768]$, followed by a pixel decoder for message extraction. At each adversarial step t, visual features are extracted with a frozen DINOv3 ViT-S/16 backbone (Siméoni et al., 2025) and projected by a two-layer MLP before being passed to the GRU controller in Eq. 9. A two-layer MLP attack head maps the hidden state to logits $\ell_t \in \mathbb{R}^K$ for Gumbel-Softmax selection. The DINOv3 backbone is frozen to avoid representation collapse, and the adversary only finetunes the GRU and MLP parameters for attack selection.

Training hyperparameters. For all methods, both the watermark model (θ, ψ) and adversary ϕ are trained on a 50,000-image subset of SA-1B (Kirillov et al., 2023) for 500 epochs with an effective batch size of 50 (batch size 25 with gradient accumulation over 2 steps) using 4 NVIDIA A4000 GPUs, yielding approximately 120K gradient steps. We use AdamW with learning rate 5×10^{-4}, a cosine schedule decaying to 10^{-6}, and 5 epochs of linear warm-up before activating the adversary. The watermark scaling factor follows a cosine decay from $\alpha_0 = 1.0$ to $\alpha_1 = 0.2$ over the final 200 epochs. Loss weights are $\lambda_{\text{dec}} = 1.0$ for message decoding and $\lambda_i = 0.1$ for imperceptibility, both applied in YUV space. Unless noted otherwise, the adversary uses depth $T \in \{1, 2\}$, entropy coefficient $\lambda_{\text{ent}} = 0.1$, Gumbel temperature $\tau = 1.0$, and hidden dimension $d_h = 384$.

Faster AutoAugment baseline. We implement Faster AutoAugment (FAA) following Hataya et al. (2020). FAA first searches an augmentation policy jointly with a 32-bit proxy watermark model for 100 epochs, then freezes the selected policy and trains the evaluated watermark model for 500 epochs. The search and training use the same attack families and parameter ranges as the corresponding CAT experiments. Thus, FAA supplies a fixed-policy comparison: its policy is selected once from the proxy objective and is not updated from the evolving watermark model or conditioned on the intermediate attacked image during training. The FAA rows in the result tables report this fixed-policy procedure without seed confidence intervals; the CLIC FAA rows use the CLIC-train split.

B.2 Entropy Regularization and Optimization Details

Entropy decomposition and single-pass optimization. Since the policy factorizes as $\pi_\phi(a) = \prod_t \pi_{\phi,t}(a_t)$, the joint entropy decomposes as $\mathcal{H}(\pi_\phi) = \sum_t \mathcal{H}(\pi_{\phi,t})$. Defining the per-step softmax distribution $p_t^{(i)} = \text{softmax}(\ell_t/\tau_{\text{ent}})_i$ and entropy $H_t = -\sum_{i=1}^K p_t^{(i)} \log p_t^{(i)}$, the adversary objective becomes

$$\mathcal{J}_{\text{adv}}(\phi) = \mathbb{E}_{a\sim\pi_\phi}\big[\mathcal{L}_{\text{msg}}\big(D_\psi(x_T),\, m\big)\big] + \lambda_{\text{ent}} \sum_{t=0}^{T-1} H_t, \tag{10}$$

$$H_t = -\sum_{i=1}^{K} p_t^{(i)} \log p_t^{(i)}. \tag{11}$$

This formulation makes the entropy bonus straightforward to compute from the controller logits at each attack step. In practice, it keeps the controller exploratory early in training, avoids premature collapse to a trivially destructive strategy, and improves robustness beyond a single dominant failure mode. We then optimize the watermark model and adversary jointly in a single pass rather than through a separate inner maximization loop. Concretely, we implement the entropy bonus as an entropy loss $\mathcal{L}_{\text{ent}} := -\sum_{t=0}^{T-1} H_t$, so the adversary receives the sign-reversed task gradient

$$\nabla_\phi \mathcal{L}_{\text{adv}} = -\nabla_\phi \mathcal{L}_{\text{msg}} + \lambda_{\text{ent}}\, \nabla_\phi \mathcal{L}_{\text{ent}}. \tag{12}$$

This lets the watermark model and the adversary share a single computational graph per iteration instead of alternating between separate inner and outer optimization loops.

B.3 Evaluation Methodology

Metrics. We evaluate robustness primarily with bit accuracy and capacity. Bit accuracy is the fraction of watermark bits recovered correctly after augmentation, computed by thresholding the extractor outputs at 0.5 and averaging the resulting bitwise matches against the ground-truth message. Capacity is reported through the corresponding binary symmetric channel estimate: for a bit recovered with accuracy p, the per-bit capacity is $1-H(p)$, where $H(p) = -p\log_2 p-(1-p)\log_2(1-p)$ is the Bernoulli entropy, and the total capacity multiplies this quantity by the payload size. We additionally compute a one-sided binomial-test p-value against the null hypothesis of random guessing ($p = 0.5$) to verify that decoding performance is statistically above chance. For perceptual distortion, image experiments report PSNR, SSIM, MS-SSIM, and LPIPS, while video experiments report VMAF and BD-rate.

Evaluation pipeline. Each sample is first watermarked by the model, producing a marked image or video together with the target message. We then apply either a single-step attack, a composed augmentation, or a compositional attack pair, depending on the experiment. The extractor is run on the transformed content, and the recovered message is compared against the original payload to compute bit accuracy, capacity, and significance statistics. For videos, framewise predictions are aggregated by averaging before thresholding, which improves stability when the watermark signal is distributed across time.

Augmentation protocol. For images, the evaluation suite covers geometric, photometric, and compression perturbations including rotation, resize, crop, perspective distortion, horizontal flip, brightness, contrast, hue, grayscale conversion, Gaussian blur, and JPEG compression. In the compositional setting, we additionally evaluate pairs of the base image augmentations at reduced strength to measure robustness under weak but composed distortions. For videos, we extend the same families with codec-based and temporal corruptions, including H.264/H.265 compression, playback-speed changes, frame dropping, temporal averaging, and temporal reordering. We also report results for stronger composed chains such as compression $\rightarrow$ crop $\rightarrow$ brightness to test robustness under realistic sequential edits.

Datasets and implementation details. Image evaluation uses SA-1B for in-distribution testing together with multiple out-of-distribution image sets (CLIC, DIV2K, and MetFace), while video evaluation is reported on Movie-Gen-Bench (Polyak et al., 2024) for in-distribution testing and SA-V (Ravi et al., 2024) for out-of-distribution testing. Unless otherwise specified, all listed augmentations are evaluated independently, and the compositional appendix results additionally include the full pairwise augmentation set. LPIPS uses the AlexNet backbone, and video quality metrics are computed with the same settings across methods to ensure a fair comparison.

Table 5: Evaluation-time augmentation grids for image and video experiments. Single-step attacks use the full parameter grid for each augmentation. Compositional attacks use pairs of augmentations and the reduced parameter grids listed here; for binary augmentations, the same toggle is used in both settings.

Domain	Augmentation	Single-step parameters	Compositional parameters
Image	Rotate	$\{5^\circ, 10^\circ, 30^\circ, 45^\circ, 90^\circ\}$	$\{5^\circ, 10^\circ\}$
Image	Resize	$\{0.32, 0.45, 0.55, 0.63, 0.71, 0.77, 0.84, 0.89, 0.95, 1.00\}$	$\{0.71, 0.77, 0.84, 0.89\}$
Image	Crop	$\{0.32, 0.45, 0.55, 0.63, 0.71, 0.77, 0.84, 0.89, 0.95, 1.00\}$	$\{0.71, 0.77, 0.84, 0.89\}$
Image	Perspective	$\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8\}$	$\{0.1, 0.2, 0.3, 0.4\}$
Image	Horizontal flip	{off, on}	—
Image	Brightness	$\{0.1, 0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 1.75, 2.0\}$	$\{0.5, 0.75, 1.25, 1.5\}$
Image	Contrast	$\{0.1, 0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 1.75, 2.0\}$	$\{0.5, 0.75, 1.25, 1.5\}$
Image	Hue	$\{-0.4, -0.3, -0.2, -0.1, 0.0, 0.1, 0.2, 0.3, 0.4, 0.5\}$	$\{-0.3, -0.2, -0.1, 0.1, 0.2, 0.3\}$
Image	Grayscale	{off, on}	—
Image	Gaussian blur	$\{3, 5, 9, 13, 17\}$ px kernel	$\{3, 5, 9\}$ px kernel
Image	JPEG	$\{40, 50, 60, 70, 80, 90\}$ quality	$\{60, 70, 80\}$ quality
Video	Rotate	$\{5^\circ, 10^\circ, 30^\circ, 45^\circ, 90^\circ\}$	$\{5^\circ, 10^\circ\}$
Video	Resize	$\{0.32, 0.45, 0.55, 0.63, 0.71, 0.77, 0.84, 0.89, 0.95, 1.00\}$	$\{0.71, 0.77, 0.84, 0.89\}$
Video	Crop	$\{0.32, 0.45, 0.55, 0.63, 0.71, 0.77, 0.84, 0.89, 0.95, 1.00\}$	$\{0.71, 0.77, 0.84, 0.89\}$
Video	Perspective	$\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8\}$	$\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6\}$
Video	Horizontal flip	{off, on}	—
Video	Brightness	$\{0.1, 0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 1.75, 2.0\}$	$\{0.5, 0.75, 1.25, 1.5\}$
Video	Contrast	$\{0.1, 0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 1.75, 2.0\}$	$\{0.5, 0.75, 1.25, 1.5\}$
Video	Hue	$\{-0.4, -0.3, -0.2, -0.1, 0.0, 0.1, 0.2, 0.3, 0.4, 0.5\}$	$\{-0.2, -0.1, 0.1, 0.2\}$
Video	Saturation	$\{0.5, 1.5\}$	$\{0.5, 0.75, 1.25, 1.5\}$
Video	Grayscale	{off, on}	—
Video	Gaussian blur	$\{3, 5, 9, 13, 17\}$ px kernel	$\{3, 5, 9\}$ px kernel
Video	JPEG	$\{40, 50, 60, 70, 80, 90\}$ quality	$\{60, 70, 80\}$ quality
Video	H.264	$\{9, 16, 23, 30, 40, 50\}$ CRF	$\{28, 34\}$ CRF
Video	H.265	$\{9, 16, 23, 30, 40, 50\}$ CRF	—
Video	Speed change	$\{0.5\times, 0.75\times, 1.25\times, 2.0\times\}$	$\{0.75\times, 1.25\times\}$
Video	Drop frame	$\{0.0, 0.15, 0.3\}$ drop prob.	$\{0.0, 0.15\}$ drop prob.
Video	Window averaging	$\{0.2, 0.4, 0.6, 0.8\}$	$\{0.2, 0.4\}$
Video	Temporal reorder	$\{0.0, 0.25, 0.5, 0.75, 1.0\}$	$\{0.25, 0.5\}$

C Dissecting CAT: Sequential Factorization, State Conditioning, and Policy Stability

These experiments isolate the design choices that make CAT effective. We compare sequential CAT with a controller that predicts a complete two-step composition jointly, remove conditioning on the evolving intermediate image, track policy entropy during training, and replace the frozen DINOv3 features with ResNet18 features. Together, the comparisons distinguish sequential factorization, intermediate-state conditioning, policy stability, and feature quality from adaptive attack selection alone.

C.1 Sequential versus joint composition prediction

Sequential CAT outperforms the image-conditioned One-shot controller. Figure 6 compares CAT with an image-conditioned One-shot controller that predicts the complete two-step attack combination jointly from the initial image, without conditioning the second operation on the intermediate image. CAT is higher on every attack-family pair. The One-shot controller's weighted overall bit accuracy and capacity are 82.7% and 61.25 bits, compared with 96.4% and 109.57 bits for CAT. The entropy trace in the same figure shows that the joint controller collapses during training, while CAT retains a diverse policy; the stability of these trajectories is analyzed separately in Section C.3.

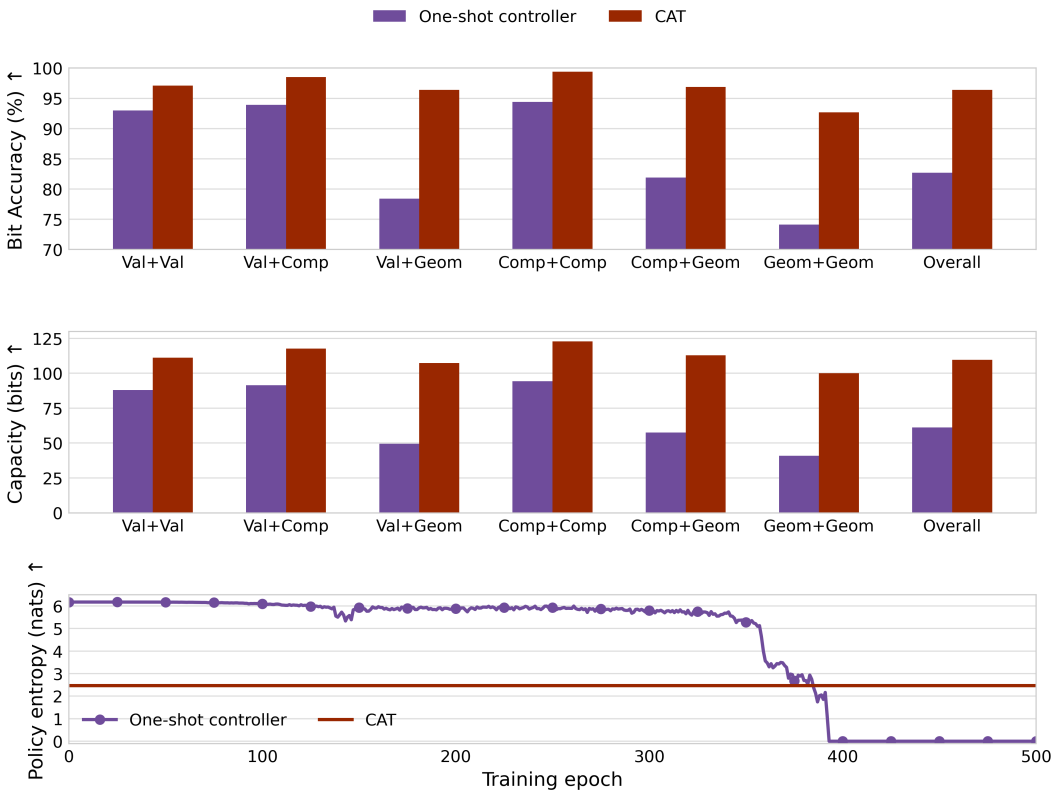


Figure 6: **CAT outperforms the image-conditioned One-shot controller across all compositional attack pairs.** Bit accuracy and capacity for the image-conditioned One-shot controller and CAT, followed by the One-shot controller's policy entropy during training on PixelSeal and SA-1B.

C.2 Intermediate-state conditioning

CAT improves the overall result over the non-image-conditioned controller. Figure 7 compares CAT with a controller that does not condition attack selection on the image. The non-image-conditioned controller is stronger on Val+Val, Val+Comp, and Comp+Comp, while CAT leads on Val+Geom, Comp+Geom, and Geom+Geom and has the stronger overall result, 96.4%/109.57 bits versus 94.3%/99.7 bits. The comparison shows that learned sequential selection alone is insufficient: conditioning each decision on the realized intermediate image adds robustness on geometric compositions and improves the aggregate result.

C.3 Policy stability and entropy dynamics

Sequential factorization avoids the policy collapse observed with one-shot composition prediction. The entropy panel in Figure 6 tracks the image-conditioned One-shot controller, which starts near 6.17 and collapses to zero by epoch 394. CAT remains flat at 2.47 nats throughout training. This behavior shows why the sequential factorization matters for optimization: the policy maintains diverse attack exploration while learning from the evolving intermediate states, whereas direct prediction over complete compositions loses its training signal through early collapse.

The controller selects attack families, while continuous attack strengths are sampled from the prescribed severity ranges. Thus, the entropy curve characterizes discrete attack-policy diversity; it is not a learned parameter-strength schedule.

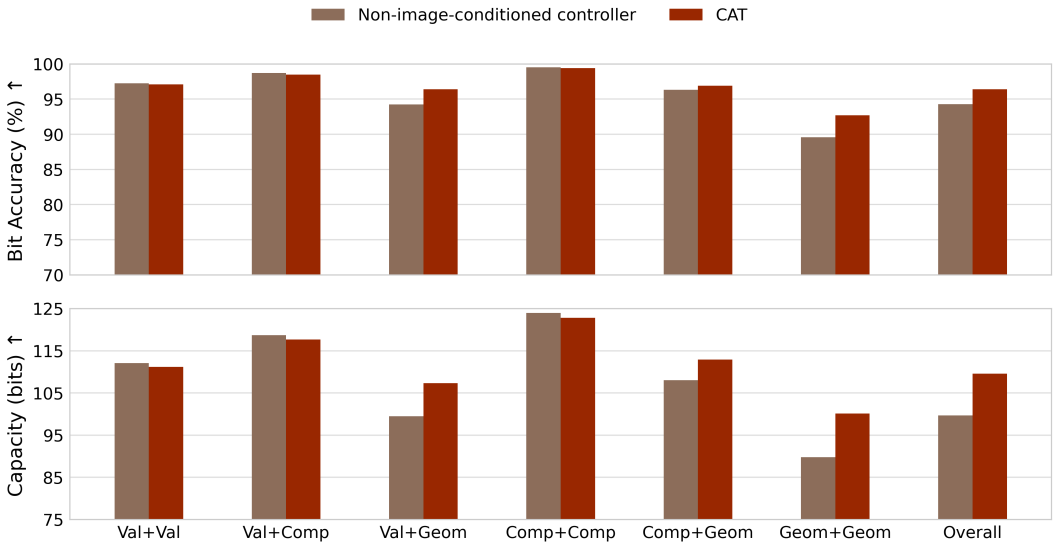


Figure 7: **CAT has the stronger overall result than the non-image-conditioned controller.** Bit accuracy and capacity for six compositional attack families and the overall result on PixelSeal and SA-1B.

C.4 Visual encoder ablation

CAT is not dependent on a particular frozen visual encoder. The ablation replaces the frozen DINOv3 features with ResNet18 features while keeping the controller and training procedure fixed. On single-step attacks, the overall result decreases from 95.6%/105.20 bits with DINOv3 to 88.4%/80.66 bits with ResNet18. On compositional attacks, bit accuracy remains similar (96.4% versus 96.5%), while capacity changes from 109.57 to 109.21 bits. The results show that the controller benefits from stronger frozen features, while the sequential policy formulation remains effective across feature backbones (Tables 3 and 4).

D Comparison with Alternative Adaptive and Adversarial Training Strategies

We compare CAT with adaptive strategies that change attack selection without using the full sequential, state-conditioned controller. These experiments test whether CAT's gains can be reproduced by a fixed curriculum, a global UCB attack-frequency policy, or a score-function policy update. The handcrafted schedule appears below, UCB appears in the main ablation tables, and existing adversarial augmentation methods are compared by their complete training costs in Appendix D.4. The bar chart uses CAT's original main-table values.

D.1 Handcrafted Difficulty Curriculum

A handcrafted schedule trails CAT in the aggregate result. Figure 8 compares CAT with a handcrafted schedule that orders attack families from easiest to hardest while sampling from every family available at the current timestep. CAT improves on all six attack-family pairs in the displayed comparison and has the stronger reported overall result, 96.4%/109.57 bits versus 94.76%/96.8 bits. The fixed order cannot track which composition the current model is failing on.

D.2 UCB Attack-Frequency Baseline

The UCB baseline adapts attack-family frequencies with a global confidence bonus but does not condition each sequential decision on the evolving attacked image. Its weighted overall bit accuracy

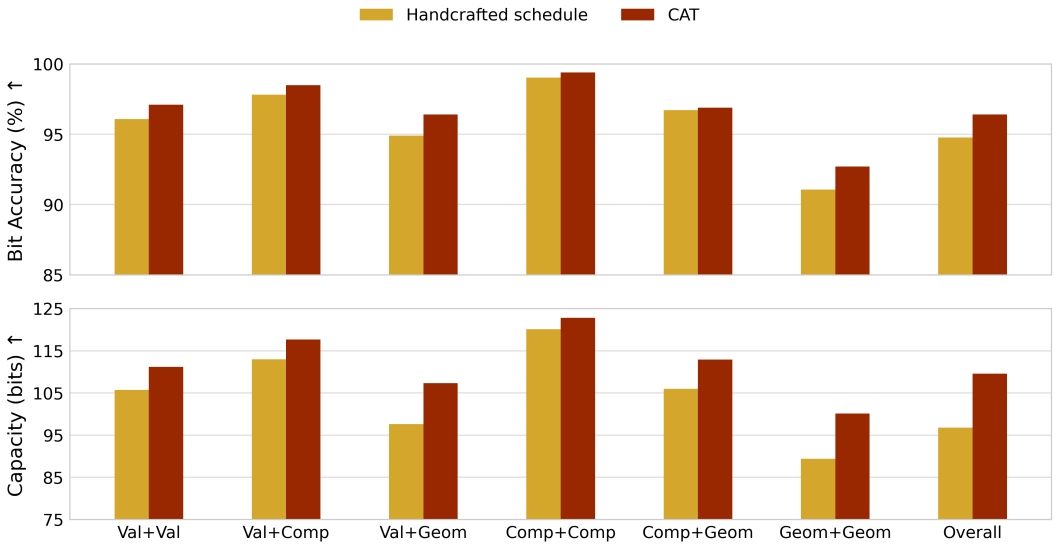


Figure 8: **CAT improves over a handcrafted easiest-to-hardest schedule.** Bit accuracy and capacity for six compositional attack families and the overall result on PixelSeal and SA-1B. The handcrafted schedule samples from all attack families available at the current timestep.

and capacity are below CAT in both the single-step and compositional ablations (Tables 3 and 4), supporting the role of intermediate-state conditioning and sequential factorization.

D.3 Reinforcement-learning controller

CAT outperforms RL on five of six attack-family pairs. Figure 9 shows that CAT exceeds RL in both metrics on five of the six pairs; RL is higher on Comp+Geom by 0.9 percentage points in bit accuracy and 0.7 bits in capacity. Relative to random augmentation, RL improves bit accuracy on four of six attack-family pairs and capacity on all six, with lower bit accuracy on Val+Val and Geom+Geom. With the controller architecture held fixed, the results favor pathwise Gumbel-Softmax updates over the tested REINFORCE-style alternative for learning a broadly effective attack policy.

D.4 Most existing adversarial augmentation methods don't scale

Cost accounting. We report the cost of the complete training and search procedures using per-step and per-epoch measurements. All times use four GPUs and include the full 500-epoch training schedule. Figure 10 shows total GPU-hours on a logarithmic scale for single-step and compositional attacks. The figure uses Fast AA for Fast AutoAugment (Lim et al., 2019), FasterAA for Faster AutoAugment (Hataya et al., 2020), AdvAA for Adversarial AutoAugment (Zhang et al., 2019b), GAT for Generalized Adversarial Training (Hsiung et al., 2023), and RoMark for the robust watermarking method of Wen and Aydore (Wen & Aydore, 2019).

Random augmentation and CAT include training costs only. Fast AA uses five folds of 100 search epochs each followed by 500 epochs of training. FasterAA uses 100 search epochs followed by 500 epochs of training. AdvAA uses $M = 8$ policy-augmented copies. RoMark's compositional cost grows exponentially with the composition depth. GAT uses $T = 7$ inner attack iterations. The Fast AA, FasterAA, and AdvAA measurements include both search and training.

Why existing methods do not scale to watermark training. Existing adversarial augmentation methods that require separate policy search, inner-loop optimization, or repeated policy evaluations are not feasible for large-scale watermark adversarial training. CAT requires 246 GPU-hours for single-step attacks and 277 GPU-hours for compositional attacks, compared with 838 and 955 GPU-hours for AdvAA, 2,988 and 2,896 GPU-hours for GAT, and 1,770 and 61,673 GPU-hours for RoMark. FasterAA is closest to CAT at 242 and 254 GPU-hours, but it still adds a separate search

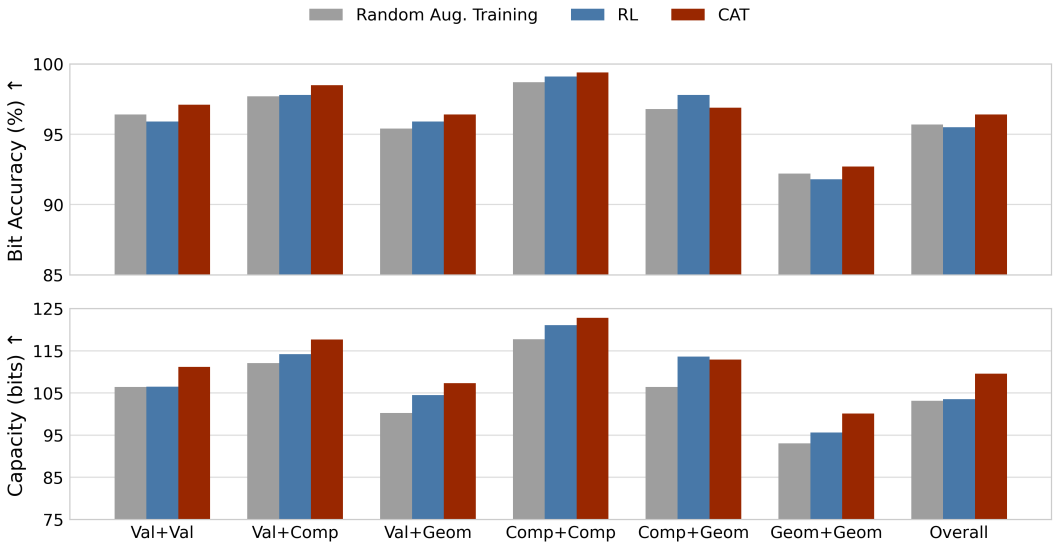


Figure 9: **CAT outperforms the RL controller on five of six attack-family pairs.** Bit accuracy and capacity for random augmentation, the RL controller, and CAT on PixelSeal and SA-1B under compositional attacks.

stage and freezes a policy learned from a proxy objective. These costs become more consequential when search must be repeated for different watermark backbones, payloads, datasets, or attack depths. CAT keeps adaptive selection inside one shared training update, avoiding the repeated decoder evaluations and separate search procedures that make these alternatives impractical at watermark scale.

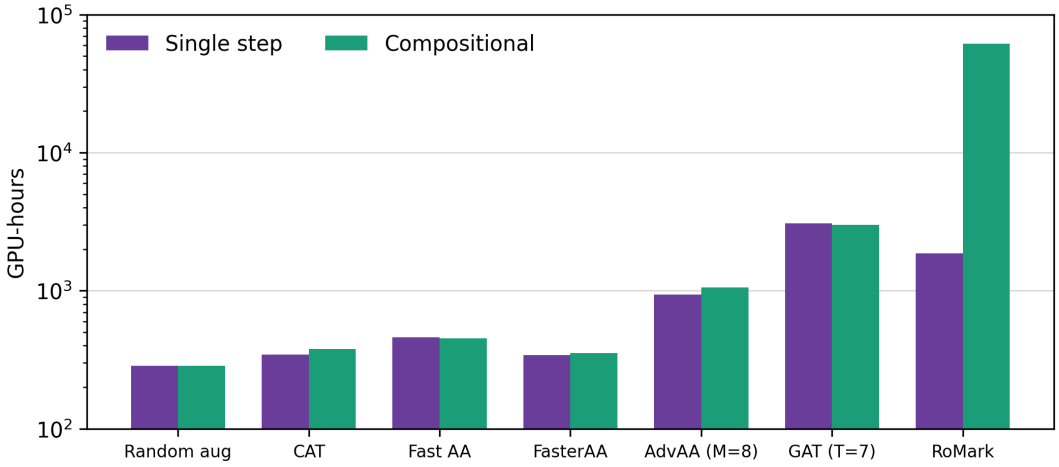


Figure 10: GPU-hour cost for the complete training and search procedures for single-step and compositional adversarial training.

E Generalization to Longer Attack Chains

E.1 Generalization beyond the training depth

We evaluate depth-2-trained models on depth-3 attack chains without retraining them at the longer depth.

CAT's advantage persists at an unseen attack depth. Figure 11 shows higher bit accuracy and capacity for CAT on all three backbones. Capacity rises from 112.45 to 121.16 for PixelSeal, 86.93 to 92.72 for VideoSeal 0.0, and 108.25 to 136.18 for VideoSeal 1.0: relative gains of 7.7%, 6.7%, and 25.8%. The largest absolute capacity gain is also on VideoSeal 1.0 (27.93 bits), whose random-augmentation baseline has the lowest bit accuracy despite its larger payload. CAT therefore provides the largest capacity gains where decoding is most difficult.

The learned robustness transfers beyond the training horizon. The additional attack is introduced only at evaluation, so these gains do not require training the controller at the longer depth or paying for deeper training. Improvement across all three backbones shows that CAT learns resistance to vulnerabilities shared by two- and three-step pipelines and transfers beyond the exact training depth.

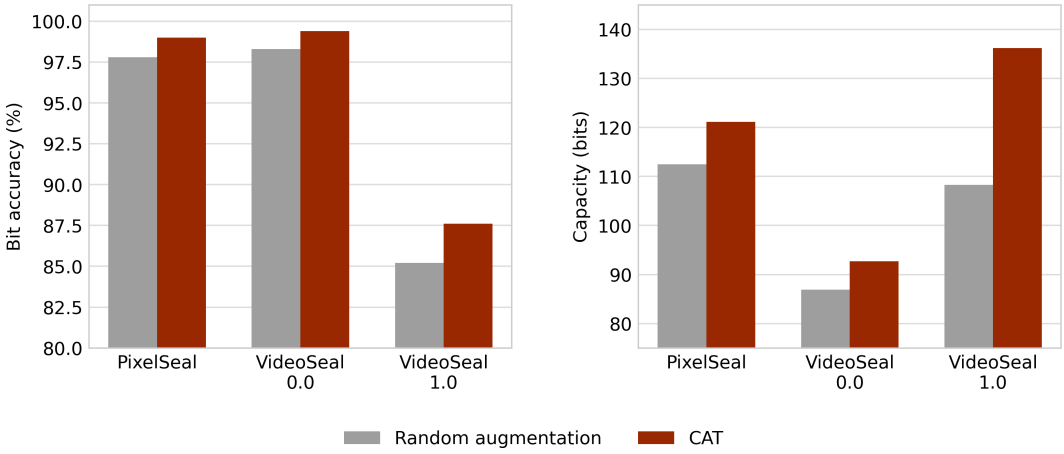


Figure 11: **Two-step CAT training improves robustness on three-step attack chains.** Bit accuracy (left) and capacity (right) for models trained at depth 2 and evaluated at depth 3. Gray/orange bars denote random augmentation/CAT. Both metrics improve for every backbone, with the largest relative capacity gain for VideoSeal 1.0. Both axes are truncated to make the differences visible.

E.2 Simulated social-media pipelines

Evaluation setup. To test robustness beyond short compositions, we evaluate fixed pipelines containing repeated transformations intended to approximate social-media reposting and light editing operations. The image suite contains five chains: L1 is repeated JPEG recompression, L2 is repeated crop–JPEG reposting, L3 adds small rotations, L4 applies repeated light contrast and compression edits, and L5 combines crop, rotation, perspective distortion, and JPEG compression. These chains contain 10–12 steps and are evaluated on 1,000 SA-1B test images. The video suite contains eight chains: LV1 is a color-regrading control, LV2–LV5 simulate repeated reposting, retiming, tilting, and screen recapture, and LV6–LV8 combine editing operations with upload, download, and re-edit steps. These chains contain 6–9 steps and are evaluated on 100 SAV-Test videos. We score every prefix of each chain and report the full-chain results below. The models are trained at $T = 1$ or $T = 2$; no long-chain training is used.

CAT transfers to simulated deployment-style pipelines. On the image chains, CAT improves the mean full-chain accuracy for every backbone and training depth. The random-augmentation VideoSeal 1.0 model trained at $T = 2$ is near chance on every image chain, while CAT substantially improves its full-chain robustness. PixelSeal and VideoSeal 0.0 also retain high accuracy across the five chains, with CAT improving their means at both depths. On video, CAT improves the mean across all six backbone/depth settings, with gains ranging from 5.2 percentage points for VideoSeal 1.0 at $T = 1$ to 25.6 points for PixelSeal at $T = 1$. The depth comparison is not monotonic: PixelSeal and VideoSeal 0.0 have higher CAT means at $T = 1$ than at $T = 2$, so the relevant comparison is CAT against its corresponding random baseline at each training depth. The results show that CAT

Table 6: Full-chain bit accuracy (%) on 1,000 SA-1B test images for five simulated social-media pipelines. The models are trained at $T = 1$ or $T = 2$ and evaluated zero-shot on the 10–12-step chains; the final column is the mean over L1–L5.

Model	T	Training	L1	L2	L3	L4	L5	Mean
PixelSeal	1	Random	91.6	79.8	62.1	80.8	63.3	75.5
	1	CAT	99.4	96.4	76.5	96.0	73.2	**88.3**
	2	Random	98.3	93.9	85.0	94.4	83.0	90.9
	2	CAT	99.8	96.7	91.3	97.1	91.3	**95.2**
VideoSeal 0.0	1	Random	99.8	96.8	76.6	96.5	77.0	89.4
	1	CAT	99.8	98.1	81.3	97.7	79.9	**91.4**
	2	Random	99.5	97.5	92.0	97.6	91.7	95.7
	2	CAT	99.9	98.3	94.4	98.4	94.7	**97.1**
VideoSeal 1.0	1	Random	89.5	81.3	59.2	82.0	58.2	74.0
	1	CAT	93.4	87.7	58.6	85.8	59.9	**77.1**
	2	Random	50.9	50.9	51.0	50.9	50.9	50.9
	2	CAT	93.6	87.8	70.5	88.8	70.1	**82.2**

Table 7: Mean full-chain bit accuracy (%) over three seeds on 100 SAV-Test videos for eight simulated social-media pipelines. Models are trained at $T = 1$ or $T = 2$ and evaluated zero-shot on the 6–9-step chains; the final column is the mean over LV1–LV8.

Model	T	Training	LV1	LV2	LV3	LV4	LV5	LV6	LV7	LV8	Mean
PixelSeal	1	Random	67.1	51.0	52.1	51.9	53.3	51.1	61.6	51.6	55.0
	1	CAT	98.9	82.7	84.9	79.6	71.4	71.0	87.4	68.9	**80.6**
	2	Random	89.6	54.1	58.0	54.1	51.4	52.1	70.6	54.6	60.6
	2	CAT	98.0	83.7	85.6	75.8	55.8	58.3	79.5	67.7	**75.5**
VideoSeal 0.0	1	Random	97.7	65.3	75.0	67.7	73.8	65.7	89.8	67.4	75.3
	1	CAT	99.6	91.8	92.9	86.2	78.4	81.3	95.4	85.3	**88.9**
	2	Random	81.8	58.9	66.4	59.8	62.1	59.0	71.4	62.8	65.3
	2	CAT	98.9	89.3	90.6	85.2	64.3	74.8	88.4	80.4	**84.0**
VideoSeal 1.0	1	Random	71.5	53.4	55.6	51.9	55.7	54.4	66.1	54.9	57.9
	1	CAT	80.4	64.1	66.5	60.0	53.4	53.4	72.9	54.5	**63.1**
	2	Random	50.5	50.6	50.7	50.6	50.4	50.6	50.7	50.7	50.6
	2	CAT	89.0	78.1	80.7	74.4	58.7	60.4	71.2	64.9	**72.2**

improves transfer across repeated transformations even when increasing the training depth does not uniformly increase the full-chain mean.

E.3 Native three-step training

Geometric-heavy compositions show the largest gains. With PixelSeal trained directly at $T = 3$, Figure 12 shows that overall bit accuracy rises from 95.3% to 97.6%. This reduces the overall bit error from 4.7% to 2.4%, a 48.9% relative reduction. The largest absolute accuracy gains occur in GGG (+8.5 percentage points), GGV (+4.5 percentage points), and CGG (+4.2 percentage points). For GGG, the hardest category under random augmentation, accuracy increases from 87.1% to 95.6%, reducing bit error by 65.9%. Thus, the overall improvement includes substantial recovery on difficult geometric-heavy compositions rather than only marginal changes on easy attacks.

The gains reflect a trade-off across attack families. CAT improves six of ten categories, while CCC, CCV, CVV, and VVV decrease by 0.5, 0.6, 0.5, and 0.2 percentage points, respectively. These four categories already exceed 98% accuracy under random training, whereas the largest gains occur where the baseline leaves substantial room for improvement. Native three-step training therefore extends CAT's benefit to a longer training horizon, with the largest gains in difficult geometric-heavy compositions.

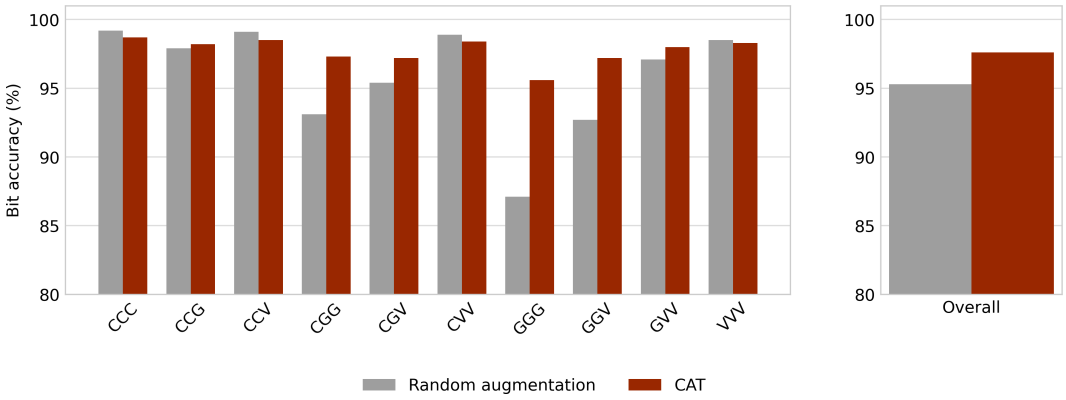


Figure 12: **Native three-step CAT training particularly benefits geometric-heavy attack compositions.** PixelSeal bit accuracy by triple category for random augmentation (gray) and CAT (orange), with the reported overall result shown separately. C, G, and V denote compression, geometric, and value transformations.

F Generalization to Realistic Removal and Editing Attacks

We evaluate zero-shot generalization to difficult removal attacks by training with the standard attack sets and testing on held-out neural compression, VAE regeneration, region inpainting, video transcoding, and diffusion regeneration. We compare random-augmentation and CAT training for VideoSeal 0.0, VideoSeal 1.0, and PixelSeal. The transcoding result is a video evaluation; the other attack categories are reported separately rather than pooled into an overall score.

CAT improves robustness to the tested compression and editing attacks. Table 8 shows improvements in both metrics for every backbone under neural compression, VAE regeneration, region inpainting, and video transcoding. For PixelSeal, capacity increases from 67.9 to 108.1 bits under neural compression, from 13.5 to 24.2 under VAE regeneration, and from 22.0 to 60.1 under transcoding (approximately $1.59\times$, $1.79\times$, and $2.73\times$). The inpainting increase from 1.0 to 3.8 bits is $3.8\times$ relatively but remains small in absolute terms.

CAT targets content-preserving transformations. Under diffusion regeneration, bit accuracy lies between 50.0% and 51.6% and capacity between 0.7 and 1.0 bits across all reported settings. As expected, neither random augmentation training nor CAT provide practically useful recovery in this experiment. This is consistent with prior theoretical results establishing fundamental limitations of pixel-level invisible watermarks under regeneration attacks, including diffusion regeneration (Zhao et al., 2024; Saberi et al., 2023). CAT instead targets robustness to content-preserving transformations for which recovery remains feasible.

Table 8: **Robustness to realistic removal and editing attacks.** Bit accuracy is reported in percent and capacity in bits. Higher is better for both metrics.

Attack	Train	VideoSeal 0.0		VideoSeal 1.0		PixelSeal	
		Bit acc. (%)	Capacity	Bit acc. (%)	Capacity	Bit acc. (%)	Capacity
Neural compression	Random	99.0	90.8	88.7	127.4	89.4	67.9
	CAT	**99.1**	**91.0**	**89.6**	**148.1**	**97.1**	**108.1**
VAE regeneration	Random	72.7	27.4	62.5	25.7	61.7	13.5
	CAT	**76.7**	**34.3**	**64.3**	**35.5**	**69.2**	**24.2**
Region inpainting	Random	56.2	5.5	53.1	4.8	50.8	1.0
	CAT	**59.8**	**9.4**	**54.2**	**8.3**	**55.0**	**3.8**
Transcoding (video)	Random	85.4	40.2	75.9	54.4	73.4	22.0
	CAT	**94.2**	**67.2**	**86.5**	**112.5**	**87.4**	**60.1**
Diffusion regeneration	Random	50.5	0.7	50.6	0.8	50.1	0.8
	CAT	**51.6**	**1.0**	50.0	0.8	**50.9**	**0.9**

G Generalization Across Watermarking Paradigms

G.1 Autoregressive watermarking setup

Training and evaluation protocol. We additionally evaluate CAT in the autoregressive image-generation setting using the WMAR framework of (Jovanović et al., 2025). Following their setup, the experiments are run on autoregressive generators based on Taming (Esser et al., 2021) and RAR-XL (Yu et al., 2025), and robustness is evaluated with the same detection metric reported in that work: true positive rate at a fixed false positive rate of 1% (TPR@FPR$= 1\%$). WMAR studies robustness under no attack, value perturbations, geometric perturbations, adversarial purification, and neural compression, and also introduces reverse cycle-consistency finetuning together with an optional synchronization layer to improve robustness after retokenization. Our tables therefore compare the non-finetuned base watermark, random-augmentation finetuning, random-augmentation finetuning with synchronization, CAT, and CAT with synchronization under the same attack families and detection metric. We supplement these aggregate benchmarks with three visual diagnostics: continuous attack sweeps in Figure 13, token-match histograms in Figure 15, and clean ROC curves in Figure 14.

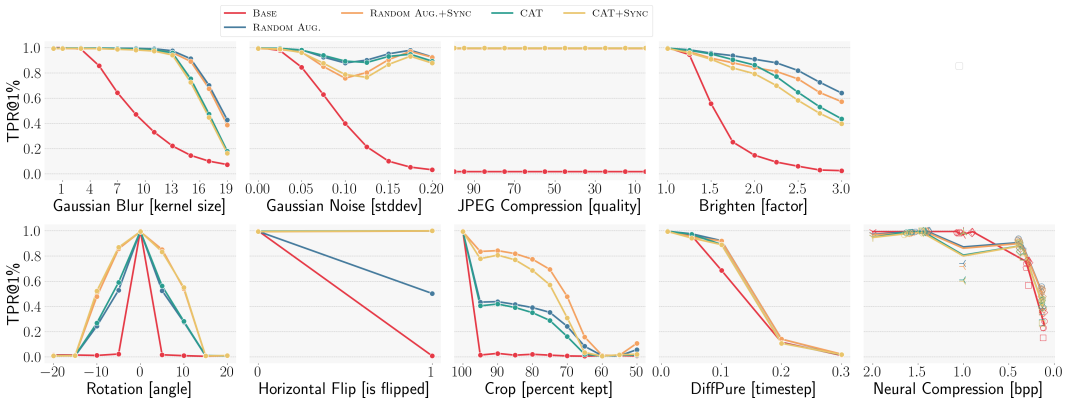
G.2 Taming and RAR-XL robustness

CAT improves the hardest geometric failure modes while maintaining performance on other attack families. Tables 9 and 10 show that the plain CAT variant consistently improves the hardest geometric failure modes while remaining competitive on the other attack families. On Taming, CAT raises geometric robustness from 0.01 to 0.52, while preserving strong performance on value perturbations (0.94), adversarial purification (0.89), and neural compression (0.87). On RAR-XL, CAT raises geometric robustness from 0.04 to 0.16 while maintaining high detection on value perturbations (0.92), adversarial purification (1.00), and neural compression (0.93).

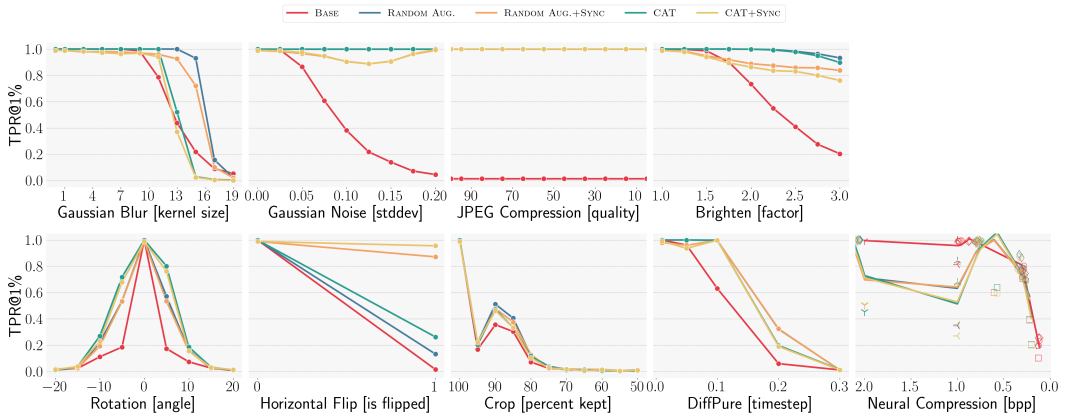
Figure 13 makes the same pattern visible continuously across attack strengths. For both backbones, the Base watermark collapses quickly under rotation, cropping, and other geometric perturbations, whereas CAT substantially widens the usable operating range. On Taming, CAT maintains much higher detection than the Base watermark across blur, noise, brighten, and rotation even before synchronization is added. On RAR-XL, CAT is likewise far more robust than the Base watermark across the geometric stress tests, with especially clear gains throughout the crop sweep. Figure 14 shows that these robustness gains preserve clean discriminability, and all methods approach near-perfect true positive rates by the operational 10^{-2} threshold used in the tables.

G.3 Synchronization and token alignment

Synchronization is what converts those gains into the strongest autoregressive robustness. The synchronization layer closes the remaining gap on geometric attacks: on both generators, CAT+Sync yields the strongest geometric robustness, reaching 0.71 on Taming and 0.38 on RAR-XL, above

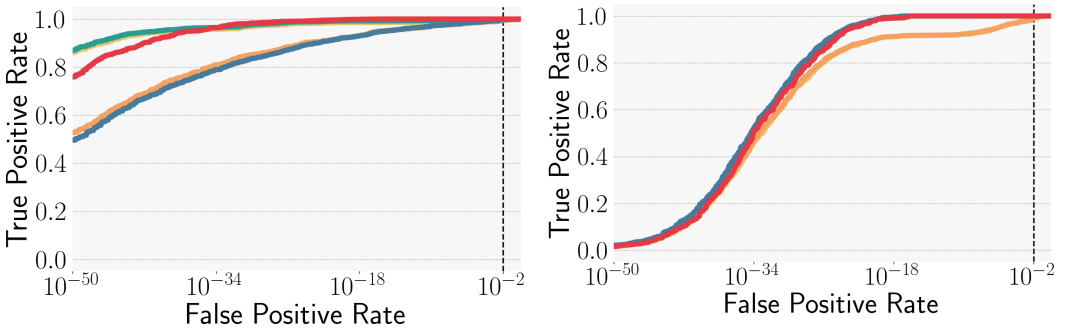


(a) Taming



(b) RAR-XL

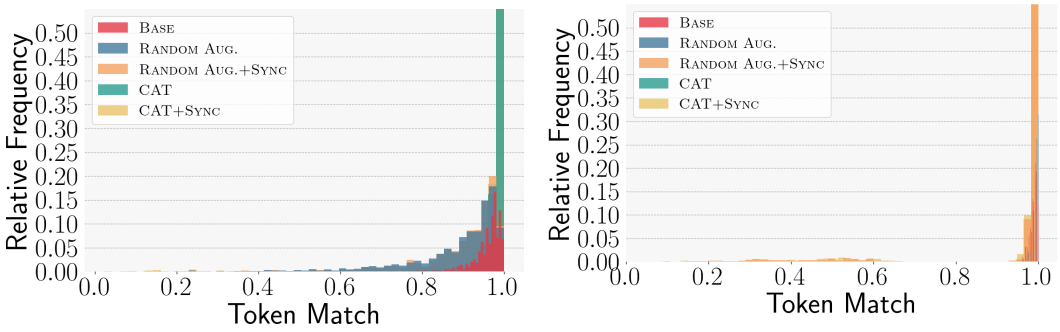
Figure 13: **Continuous attack sweeps for autoregressive watermark robustness.** Each panel plots TPR@FPR$= 1\%$ as the corruption strength varies. CAT consistently enlarges the robustness envelope over the Base watermark, while synchronization is particularly helpful for alignment-disrupting transformations such as cropping and horizontal flips.



(a) Taming (b) RAR-XL

Figure 14: **ROC curves for clean autoregressive watermark detection.** The dashed vertical line marks the operational false positive rate of 10^{-2} used in Tables 9 and 10. Extreme low-FPR behavior depends on the backbone, but all methods approach near-perfect TPR at the operating point.

both plain CAT and the corresponding random-augmentation baselines. Figure 13 shows where this extra gain comes from. On Taming, the synchronized variants are the only methods that remain reliable deep into the crop sweep, and they preserve perfect performance under horizontal flips. On



(a) Taming (b) RAR-XL

Figure 15: **Token-match histograms under clean evaluation.** Synchronization broadens the high-match distribution and introduces a modest left tail, consistent with a detector that tolerates mild token misalignment after retokenization instead of requiring exact positional correspondence.

RAR-XL, CAT+Sync is especially strong under aggressive cropping, clearly outperforming Random Aug.+Sync once the kept area becomes small, while both synchronized variants remain near-perfect under flips.

The token-match histograms in Figure 15 visualize the mechanism behind this geometric resilience. Synchronization replaces a rigid concentration at almost-perfect token match with a broader high-match distribution and a mild left tail, most visibly on RAR-XL. This pattern suggests a decoder that tolerates small alignment mismatches instead of demanding exact token-wise correspondence after retokenization. At the same time, CAT without synchronization remains sharply concentrated near perfect token match on Taming but still lacks the crop and flip robustness of CAT+Sync, showing that stronger token identities alone are not enough; the model also needs a way to realign them after spatial shifts. Taken together, these results suggest that in the autoregressive setting CAT is most useful for hardening the token representations themselves, while synchronization is essential for converting that adversarial pressure into a watermark that stays decodable after geometric misalignment.

Table 9: Autoregressive watermarking results for Taming-style generation, building on the WMAR framework of (Jovanović et al., 2025). Each entry reports TPR@FPR$= 1\%$ under no attack, value perturbations, geometric perturbations, adversarial purification, and neural compression. The best value in each column is bolded. Higher is better.

Method	None	Value	Geometric	Adv. Purif.	Neural Comp.
Base	**1.00**	0.26	0.01	0.69	0.71
Random Aug.	**1.00**	**0.99**	0.34	0.87	0.82
Random Aug.+Sync	0.99	0.89	0.62	0.88	0.81
CAT	**1.00**	0.94	0.52	**0.89**	**0.87**
CAT+Sync	0.99	0.89	**0.71**	**0.89**	**0.87**

Table 10: Autoregressive watermarking results for RAR-XL generation, building on the WMAR framework of (Jovanović et al., 2025). Each entry reports TPR@FPR$= 1\%$ under no attack, value perturbations, geometric perturbations, adversarial purification, and neural compression. The best value in each column is bolded. Higher is better.

Method	None	Value	Geometric	Adv. Purif.	Neural Comp.
Base	**1.00**	0.53	0.04	0.63	0.77
Random Aug.	0.99	**1.00**	0.11	**1.00**	0.92
Random Aug.+Sync	0.99	0.94	0.35	**1.00**	0.92
CAT	**1.00**	0.92	0.16	**1.00**	0.93
CAT+Sync	**1.00**	0.93	**0.38**	**1.00**	**0.93**

H Additional OOD Image Results

CAT's robustness gains persist under distribution shift. Tables 11 and 12 extend the main results to additional OOD datasets, namely DIV2K (Agustsson & Timofte, 2017) and MetFace. The same overall pattern remains visible: CAT improves overall robustness relative to the random-augmentation baseline trained for the same number of gradient updates and the same augmentation budget across both datasets, with the clearest gains again appearing in watermark capacity. In the single-step setting, overall capacity rises from 110.62 to 129.0 on DIV2K and from 121.84 to 140.6 on MetFace for VideoSeal 1.0, while PixelSeal improves from 59.73 to 93.63 on DIV2K and from 67.91 to 105.45 on MetFace. These gains are much larger than the corresponding changes in average bit accuracy, indicating that CAT is making decoding reliably correct across a broader set of corruptions rather than producing isolated improvements on a few easy cases.

The gains are strongest on the hardest OOD corruption settings. The hardest OOD settings again show the largest benefits. In the single-step results, PixelSeal gains 56.8% capacity on DIV2K and 55.3% on MetFace, while VideoSeal 1.0 gains 16.6% and 15.4%, respectively. In the compositional OOD results, the same pattern holds on the harder mixed corruption pairs: on DIV2K, VideoSeal 1.0 improves from 130.74 to 161.4 on Comp+Comp and from 103.87 to 131.6 on Comp+Geom, while PixelSeal improves from 116.91 to 121.1 and from 105.55 to 111.2 on the same two settings. On MetFace, VideoSeal 1.0 improves from 138.93 to 174.3 on Comp+Comp and from 122.22 to 159.0 on Comp+Geom. This behavior is consistent with the main paper's hypothesis that adaptive compositional adversaries are most valuable when robustness depends on rare, structured corruptions rather than on isolated perturbations.

OOD generalization remains consistent with the main results. Across the shifted image distributions and corruption patterns, the learned adversary improves overall robustness for all three learned backbones in both the single-step and compositional settings. The largest gains again appear for the more vulnerable backbones, especially VideoSeal 1.0 and PixelSeal, showing that targeted compositional training improves robustness beyond the training distribution.

Table 11: **The robustness gains from a single learned attack step persist across more challenging out-of-distribution datasets, with the largest improvements again appearing for VideoSeal 1.0 and PixelSeal.** Additional single-step attack results on OOD DIV2K and MetFace images. Each cell reports bit accuracy and capacity for identity, value, compression, geometric, combined, and overall evaluation settings.

		Identity		Value		Compression		Geometric		Combined		Overall	
	Model (bits)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)
DIV2k	InvisMark (100)	99.0	96.03	87.8	73.49	95.3	88.43	82.8	63.40	83.4	63.64	86.0	69.68
	TrustMark (100)	99.1	97.80	95.1	85.26	89.3	77.98	75.1	49.54	98.7	96.54	88.4	74.97
	MBRS (256)	98.2	238.04	91.1	181.85	87.8	184.94	65.1	77.28	94.4	204.49	82.6	152.37
	VideoSeal 0.0 (96)	99.3	92.24	97.6	86.31	97.2	83.96	93.4	72.17	98.7	89.56	96.4	82.02
	+ FAA	99.6	93.74	97.2	85.34	83.0	61.60	61.3	21.39	99.2	91.65	74.2	44.85
	+ CAT	$99.6^{(0.0)}$	$93.7^{(0.2)}$	$96.5^{(0.9)}$	$84.5^{(2.0)}$	$99.5^{(0.1)}$	$93.1^{(0.4)}$	$94.6^{(0.1)}$	$76.9^{(0.3)}$	$99.3^{(0.2)}$	$91.7^{(0.9)}$	$\mathbf{96.9}^{(0.3)}_{\uparrow 9.5\%}$	$\mathbf{84.77}^{(0.9)}_{\uparrow 9.4\%}$
	VideoSeal 1.0 (256)	89.3	131.86	87.3	120.14	86.4	114.63	81.3	90.08	88.4	125.72	85.4	110.62
	+ FAA	99.6	250.67	96.3	219.04	82.6	159.99	60.9	54.07	97.7	229.39	73.7	114.76
	+ CAT	$93.0^{(0.8)}$	$164.2^{(8.1)}$	$87.1^{(3.3)}$	$133.3^{(9.8)}$	$92.6^{(0.8)}$	$161.8^{(7.6)}$	$78.3^{(0.4)}$	$99.6^{(3.6)}$	$90.7^{(1.2)}$	$147.9^{(9.6)}$	$\mathbf{85.7}^{(1.0)}_{\uparrow 9.4\%}$	$\mathbf{129.0}^{(5.3)}_{\uparrow 16.6\%}$
	PixelSeal (128)	91.4	74.84	89.0	67.21	87.3	62.00	80.9	46.05	89.7	68.60	86.2	59.73
	+ FAA	94.0	89.81	88.1	69.43	77.7	53.04	59.3	17.92	89.6	72.36	69.9	37.26
	+ CAT	$97.5^{(0.2)}$	$113.4^{(0.4)}$	$93.3^{(0.3)}$	$95.9^{(1.1)}$	$96.3^{(0.5)}$	$110.1^{(1.5)}$	$84.3^{(1.6)}$	$76.7^{(5.9)}$	$96.3^{(0.3)}$	$105.9^{(2.3)}$	$\mathbf{91.3}^{(0.6)}_{\uparrow 6.0\%}$	$\mathbf{93.63}^{(2.6)}_{\uparrow 56.8\%}$
	InvisMark (100)	99.0	96.04	88.7	76.05	93.2	82.01	83.1	64.16	67.1	37.40	82.2	63.52
	TrustMark (100)	100.0	99.98	97.9	93.44	90.4	80.36	76.2	51.84	99.6	98.36	90.0	78.83
	MBRS (256)	99.5	250.90	95.1	216.87	89.1	196.88	65.6	81.63	93.0	199.79	83.8	164.20
	VideoSeal 0.0 (96)	99.9	95.54	99.6	93.85	98.9	90.87	96.6	83.45	97.2	86.33	97.9	88.11
	+ FAA	99.9	95.65	97.8	88.60	83.0	62.43	60.7	19.97	98.9	90.75	74.0	44.82
	+ CAT	$100.0^{(0.0)}$	$95.9^{(0.0)}$	$99.0^{(0.5)}$	$92.1^{(1.6)}$	$99.9^{(0.0)}$	$95.7^{(0.2)}$	$97.7^{(0.5)}$	$85.6^{(0.3)}$	$97.6^{(1.1)}$	$88.2^{(2.7)}$	$\mathbf{98.3}^{(0.5)}_{\uparrow 9.4\%}$	$\mathbf{89.4}^{(1.2)}_{\uparrow 1.5\%}$
	VideoSeal 1.0 (256)	90.4	139.36	89.7	134.84	88.6	128.26	84.8	108.87	86.9	121.09	87.2	121.84
	+ FAA	99.9	254.16	96.9	224.69	82.6	159.28	60.9	50.04	97.1	224.50	73.8	113.27
	+ CAT	$94.3^{(1.1)}$	$176.0^{(11.5)}$	$91.1^{(2.4)}$	$154.8^{(3.6)}$	$94.4^{(0.5)}$	$176.6^{(7.6)}$	$81.2^{(0.7)}$	$116.3^{(6.4)}$	$87.4^{(2.3)}$	$138.9^{(13.5)}$	$\mathbf{87.3}^{(0.8)}_{\uparrow 0.1\%}$	$\mathbf{140.6}^{(5.9)}_{\uparrow 15.4\%}$
	PixelSeal (128)	92.8	80.54	91.9	77.52	90.2	71.69	85.8	59.19	87.7	66.62	88.6	67.91
	+ FAA	95.8	99.05	89.5	76.86	78.2	55.68	59.1	16.48	86.8	67.27	70.0	38.21
	+ CAT	$99.8^{(0.2)}$	$126.4^{(1.9)}$	$97.2^{(0.6)}$	$115.3^{(3.5)}$	$99.3^{(0.4)}$	$123.9^{(2.8)}$	$88.3^{(1.0)}$	$91.3^{(3.8)}$	$95.5^{(0.7)}$	$103.9^{(8.8)}$	$\mathbf{94.0}^{(0.6)}_{\uparrow 1.1\%}$	$\mathbf{105.45}^{(5.2)}_{\uparrow 55.3\%}$

I Additional Video Results

Video training and evaluation protocol. For the additional video results, we initialize from the pretrained image watermarking model and finetune on video data from Movie-Gen-Bench. We then evaluate the resulting video model on 100 evaluation videos from Movie-Gen-Bench for in-distribution testing and 100 evaluation videos from SA-V for out-of-distribution testing. This protocol

Table 12: **Under compositional attacks, CAT yields its clearest OOD gains, with especially large improvements for VideoSeal 1.0 and PixelSeal on hard mixed and repeated attack pairs.** Additional out-of-distribution image results on DIV2K and MetFace for two-step attack sequences. Each cell reports bit accuracy and capacity for pairwise compositions of value, compression, and geometric augmentation families, plus the overall average.

		Val+Val		Val+Comp		Val+Geom		Comp+Comp		Comp+Geom		Geom+Geom		Overall	
	Model (bits)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)
DIV2k	InvisMark (100)	90.5	73.01	65.2	27.04	87.8	69.35	47.0	0.88	60.0	21.51	88.9	70.78	78.7	53.01
	TrustMark (100)	95.4	85.92	96.1	87.50	78.2	52.14	96.5	88.47	77.1	49.21	70.3	35.58	81.8	58.89
	MBRS (256)	91.4	179.76	82.5	123.78	59.8	38.69	75.2	79.13	55.0	13.86	49.6	0.97	64.6	57.37
	VideoSeal 0.0 (96)	97.4	84.53	98.5	88.28	96.2	78.64	99.1	90.92	97.2	82.37	92.8	72.38	96.4	80.74
	+ FAA	99.9	95.44	98.0	89.40	83.5	64.06	61.6	22.82	99.7	94.52	74.7	47.07	88.7	73.54
	+ CAT	$98.7^{(0.2)}$	$89.7^{(1.1)}$	$99.3^{(0.2)}$	$92.2^{(1.0)}$	$97.2^{(0.8)}$	$83.8^{(3.0)}$	$99.7^{(0.1)}$	$94.4^{(0.8)}$	$98.3^{(0.8)}$	$88.1^{(3.1)}$	$95.9^{(1.9)}$	$80.5^{(5.2)}$	**97.7**$^{(0.8)}_{+11.4\%}$	**86.24**$^{(2.8)}_{+6.8\%}$
	VideoSeal 1.0 (256)	86.9	118.04	88.1	124.43	81.9	91.54	89.1	130.74	84.8	103.87	78.9	82.20	83.6	101.16
	+ FAA	68.2	29.16	67.6	28.00	62.1	21.23	54.0	10.69	68.2	29.09	58.9	16.95	64.1	23.71
	+ CAT	$88.0^{(4.3)}$	$134.2^{(17.3)}$	$90.4^{(2.0)}$	$147.7^{(9.1)}$	$84.9^{(3.0)}$	$115.4^{(10.2)}$	$92.6^{(0.8)}$	$161.4^{(7.7)}$	$88.2^{(0.7)}$	$131.6^{(4.5)}$	$85.0^{(3.5)}$	$108.7^{(3.9)}$	**86.9**$^{(2.4)}_{+1.0\%}$	**125.38**$^{(8.5)}_{+23.9\%}$
	PixelSeal (128)	96.6	107.50	97.7	112.44	95.5	100.71	98.5	116.91	96.6	105.55	91.9	92.30	95.7	103.18
	+ FAA	86.4	56.54	83.8	50.47	73.8	36.91	57.6	12.82	86.0	55.37	67.2	26.76	77.6	42.48
	+ CAT	$97.5^{(0.7)}$	$111.8^{(4.1)}$	$98.3^{(0.6)}$	$116.5^{(4.2)}$	$95.6^{(1.2)}$	$105.0^{(2.5)}$	$99.1^{(0.5)}$	$121.1^{(3.9)}$	$96.0^{(1.8)}$	$111.2^{(3.3)}$	$95.7^{(0.4)}$	$105.2^{(1.2)}$	**96.4**$^{(0.9)}_{+1.8\%}$	**109.05**$^{(2.9)}_{+5.7\%}$
MetFace	InvisMark (100)	94.8	84.25	67.5	31.31	90.5	75.43	48.2	0.96	61.0	24.17	89.9	73.69	80.9	58.09
	TrustMark (100)	98.9	96.23	98.9	96.24	80.9	58.60	99.4	97.66	79.8	55.76	72.8	40.72	84.5	65.98
	MBRS (256)	96.4	223.33	90.2	177.38	62.5	58.61	85.3	140.42	57.6	27.29	49.6	0.96	68.0	81.60
	VideoSeal 0.0 (96)	99.0	92.59	99.8	94.88	99.0	90.81	99.9	95.33	99.3	91.85	95.8	83.70	98.7	90.80
	+ FAA	100.0	95.86	97.8	89.41	83.6	63.74	61.6	21.33	99.5	93.66	74.6	46.13	88.7	73.13
	+ CAT	$99.7^{(0.1)}$	$94.7^{(0.3)}$	$99.9^{(0.0)}$	$95.4^{(0.1)}$	$99.3^{(0.1)}$	$92.8^{(0.7)}$	$100.0^{(0.0)}$	$95.9^{(0.1)}$	$99.7^{(0.1)}$	$94.5^{(0.6)}$	$98.9^{(0.4)}$	$90.9^{(1.5)}$	**99.5**$^{(0.1)}_{+0.8\%}$	**93.49**$^{(0.6)}_{+3.0\%}$
	VideoSeal 1.0 (256)	89.5	133.85	90.0	136.70	85.6	111.78	90.3	138.93	87.9	122.22	82.2	98.89	86.7	118.51
	+ FAA	68.9	30.97	68.0	29.23	62.0	21.28	53.2	9.48	68.4	30.00	58.5	16.56	64.1	24.20
	+ CAT	$91.8^{(2.0)}$	$160.5^{(6.8)}$	$93.0^{(0.7)}$	$167.5^{(5.3)}$	$90.0^{(1.7)}$	$147.0^{(4.7)}$	$94.1^{(1.0)}$	$174.3^{(10.5)}$	$92.2^{(1.0)}$	$159.0^{(9.9)}$	$88.8^{(0.5)}$	$139.0^{(3.6)}$	**91.0**$^{(1.0)}_{+5.0\%}$	**153.19**$^{(4.1)}_{+29.3\%}$
	PixelSeal (128)	98.9	122.16	99.6	125.05	98.8	119.42	99.8	125.92	99.0	120.62	95.5	109.48	98.5	119.38
	+ FAA	86.6	57.24	84.3	52.10	73.7	36.82	57.1	11.72	85.7	54.68	66.9	26.42	77.6	42.57
	+ CAT	$99.5^{(0.0)}$	$124.5^{(0.2)}$	$99.7^{(0.0)}$	$126.0^{(0.4)}$	$98.3^{(0.4)}$	$116.9^{(2.7)}$	$99.9^{(0.1)}$	$127.2^{(0.5)}$	$99.1^{(0.8)}$	$119.8^{(3.3)}$	$97.3^{(0.0)}$	$115.2^{(2.4)}$	**98.7**$^{(0.4)}_{+0.2\%}$	**119.62**$^{(2.0)}_{+0.2\%}$

Table 13: **Video watermarking single-step attack results on in-distribution Movie-Gen-Bench (ID) and out-of-distribution SAV-Test (OOD).** Each cell reports bit accuracy and capacity for identity, value, compression, geometric, video-temporal, and combined augmentation families.

		Identity		Value		Compression		Geometric		Temporal		Combined		Overall	
	Model (bits)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)
Movie-Gen-Bench (ID)	RivaGAN (32)	88.9	20.36	69.1	9.42	72.6	9.86	74.7	10.39	80.9	14.43	51.1	1.27	70.3	9.37
	VideoSeal 0.0 (96)	95.1	69.97	93.0	62.37	84.3	45.80	94.2	67.88	90.5	59.74	72.4	16.74	88.1	53.14
	+ FAA	95.4	72.91	93.1	66.51	64.9	20.56	51.1	3.18	77.0	42.52	50.8	2.33	67.8	27.52
	+ CAT	95.2	70.04	93.4	63.67	91.5	59.00	94.6	68.95	94.1	67.57	87.0	44.80	**92.5**$_{+6.9\%}$	**61.99**$_{+16.7\%}$
	VideoSeal 1.0 (256)	86.1	107.64	82.3	87.28	74.2	65.12	84.8	103.34	80.9	90.11	60.1	9.66	77.7	75.60
	+ FAA	98.1	227.64	92.7	177.05	60.4	35.09	51.0	7.32	76.4	116.89	50.4	5.68	66.3	68.98
	+ CAT	84.9	100.44	82.9	89.90	82.8	89.43	84.3	98.74	84.7	100.01	79.2	69.22	**83.0**$_{+1.8\%}$	**90.63**$_{+19.9\%}$
	PixelSeal (128)	80.5	37.54	71.9	20.34	68.3	19.89	78.9	34.95	75.4	30.59	56.1	2.60	71.3	23.25
	+ FAA	90.7	74.04	86.9	62.68	58.3	11.41	51.3	3.23	72.6	39.28	50.3	2.20	64.0	23.68
	+ CAT	80.6	37.84	72.8	21.93	72.6	23.98	79.6	36.62	77.8	33.62	65.1	10.65	**74.3**$_{+4.2\%}$	**26.63**$_{+14.5\%}$
SAV-Test (OOD)	RivaGAN (32)	94.1	25.13	74.0	13.02	77.6	13.66	81.2	14.94	86.8	19.17	50.2	2.05	74.8	12.97
	VideoSeal 0.0 (96)	94.5	68.05	91.4	57.84	83.4	43.75	93.3	65.09	89.7	57.69	71.6	16.33	87.1	50.65
	+ FAA	94.6	69.14	92.5	63.97	64.1	19.63	50.2	2.72	76.3	40.44	50.1	1.71	67.0	26.18
	+ CAT	95.2	70.46	90.8	57.43	88.9	53.14	92.3	63.18	92.6	63.30	79.4	29.14	**89.3**$_{+2.4\%}$	**54.61**$_{+7.4\%}$
	VideoSeal 1.0 (256)	85.1	101.68	82.3	87.24	73.9	63.77	84.2	100.10	80.4	87.90	59.9	9.54	77.4	74.04
	+ FAA	96.4	205.79	91.7	169.02	60.4	35.05	50.9	7.33	75.8	109.72	50.2	5.71	66.0	65.76
	+ CAT	84.1	95.59	82.7	89.13	80.8	82.13	83.4	94.25	83.2	93.66	75.2	55.46	**81.4**$_{+5.2\%}$	**84.53**$_{+14.2\%}$
	PixelSeal (128)	79.8	36.26	68.7	16.19	67.1	17.79	76.3	29.82	73.3	26.26	52.5	0.95	68.6	19.49
	+ FAA	91.6	78.60	88.0	67.10	58.1	12.67	51.0	3.45	73.1	41.80	49.2	2.88	64.1	25.39
	+ CAT	79.8	36.10	69.1	16.93	70.4	20.66	76.5	30.26	75.0	28.00	55.2	2.45	**70.0**$_{+2.0\%}$	**20.71**$_{+6.2\%}$

Table 14: **Under compositional video attacks, CAT delivers its strongest gains on the hardest combined and temporal attack pairs, with especially large improvements for PixelSeal.** Compositional video watermarking results on in-distribution Movie-Gen-Bench (ID) and out-of-distribution SA-V (OOD). Each cell reports bit accuracy and capacity for pairwise augmentation-family compositions, including combined and temporally structured pairs.

		Val+Val		Val+Geom		Comp+Comp		Comp+Geom		Geom+Geom		Geom+Temp		Temp+Temp		Overall	
	Model (bits)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)	Bit acc. (%) (↑)	Capacity (↑)
Movie-Gen-Bench (ID)	RivaGAN (32)	70.1	8.57	74.0	10.03	71.5	9.03	74.0	10.01	77.2	11.05	82.0	14.54	87.3	18.98	78.1	12.54
	VideoSeal 0.0 (96)	94.7	69.20	97.8	82.26	82.5	40.55	97.5	81.26	96.1	76.52	97.8	82.34	90.4	60.87	96.0	76.60
	+ FAA	96.4	81.24	51.8	1.97	72.1	34.36	51.3	1.96	54.0	5.91	51.9	1.98	98.6	87.77	65.8	25.87
	+ CAT	93.7	66.09	97.3	80.01	93.7	65.17	97.4	80.32	96.0	75.96	97.6	81.39	96.0	74.65	**96.7**$_{+0.8\%}$	**77.85**$_{+1.6\%}$
	VideoSeal 1.0 (256)	82.3	87.28	86.9	113.11	65.8	33.70	86.7	112.26	83.9	99.39	86.9	113.42	74.9	66.80	84.0	101.26
	+ FAA	66.9	26.72	49.8	6.43	58.3	14.49	50.2	6.38	50.9	7.14	49.8	6.48	67.5	27.81	55.3	12.40
	+ CAT	82.9	89.90	86.0	107.49	80.9	78.77	85.7	105.42	83.5	95.19	85.9	106.48	83.5	93.55	**85.0**$_{+1.1\%}$	**101.69**$_{+0.4\%}$
	PixelSeal (128)	92.6	82.75	97.9	111.32	68.4	26.08	97.4	108.33	95.1	100.14	98.1	112.26	81.2	64.48	94.2	99.26
	+ FAA	83.4	48.88	50.1	2.27	65.1	20.18	50.5	2.27	52.2	4.31	50.0	2.27	85.8	54.15	60.5	16.04
	+ CAT	92.9	85.54	97.8	111.36	93.5	87.67	97.5	109.22	95.9	102.94	98.2	113.45	96.4	103.51	**97.0**$_{+0.9\%}$	**107.16**$_{+6.0\%}$
SAV-Test (OOD)	RivaGAN (32)	75.7	12.04	80.4	14.42	76.5	12.57	79.1	13.93	84.4	16.31	88.6	19.91	92.9	23.94	84.3	17.22
	VideoSeal 0.0 (96)	92.1	61.62	96.0	75.47	77.7	32.79	95.7	74.38	93.9	69.05	95.9	75.16	86.6	53.00	93.7	69.39
	+ FAA	96.0	79.73	50.3	2.43	70.6	32.89	49.8	2.51	52.3	5.92	50.3	2.40	98.3	86.19	64.4	25.33
	+ CAT	90.5	56.81	95.8	73.96	87.7	51.74	95.3	72.46	93.5	67.30	95.6	73.34	92.0	63.76	**94.3**$_{+1.2\%}$	**69.67**$_{+0.4\%}$
	VideoSeal 1.0 (256)	82.3	87.24	86.4	110.45	65.6	33.32	86.2	109.44	83.4	95.91	86.4	110.56	74.4	65.24	83.6	**98.84**
	+ FAA	67.7	28.80	50.5	5.38	58.8	15.14	50.8	5.73	51.5	6.66	50.6	5.32	68.0	29.47	55.9	12.32
	+ CAT	82.7	89.13	85.0	101.99	78.2	69.44	84.8	101.16	82.7	91.11	84.9	101.94	81.4	85.39	**83.9**$_{+0.4\%}$	96.64$_{-2.2\%}$
	PixelSeal (128)	83.2	53.45	91.5	83.98	65.3	20.14	90.7	80.49	87.5	70.21	91.4	83.08	77.3	50.66	87.7	73.37
	+ FAA	83.1	48.27	48.0	2.73	64.0	20.03	48.5	2.63	50.4	4.52	48.0	2.71	85.9	54.68	59.1	16.26
	+ CAT	89.7	73.02	95.6	98.80	85.5	64.09	95.1	96.65	92.8	88.32	95.7	99.07	90.8	83.58	**93.9**$_{+7.1\%}$	**92.62**$_{+26.2\%}$

isolates the effect of transferring the image-trained model into the video setting while keeping the downstream robustness evaluation fixed across the two benchmarks. As a baseline, we add RivaGAN, which is a Generative Adversarial Network based watermarking method (Zhang et al., 2019a).

CAT is most helpful on the hardest compositional and temporal video corruptions. Tables 13 and 14 extend the main results to video watermarking on in-distribution Movie-Gen-Bench (Polyak et al., 2024) and out-of-distribution SA-V (Ravi et al., 2024). In the single-step setting, CAT improves overall capacity for all three learned backbones on both datasets: on Movie-Gen-Bench, overall capacity rises from 53.14 to 61.99 for VideoSeal 0.0, from 75.60 to 90.63 for VideoSeal 1.0, and from 23.25 to 26.63 for PixelSeal; on SA-V, it rises from 50.65 to 54.61, from 74.04 to 84.53, and from 19.49 to 20.71, respectively. In the compositional setting, the largest gains concentrate on the hardest combined and temporal compositions. For example, on Movie-Gen-Bench, VideoSeal 1.0 improves from 33.70 to 78.77 on Comp+Comp and from 66.80 to 93.55 on Temp+Temp, while PixelSeal improves from 26.08 to 87.67 on Comp+Comp and from 64.48 to 103.51 on Temp+Temp.

Compositional OOD gains concentrate on hard video attacks. PixelSeal shows strong improvements across both ID and OOD video evaluation, with especially large gains in the harder composed settings. In the compositional OOD results on SA-V, PixelSeal improves from 73.37 to 92.62 overall capacity, with large gains on Comp+Comp (20.14 to 64.09), Comp+Geom (80.49 to 96.65), and Temp+Temp (50.66 to 83.58). VideoSeal 0.0 improves from 32.79 to 51.74 on Comp+Comp and from 53.00 to 63.76 on Temp+Temp. VideoSeal 1.0 improves hard regimes such as Comp+Comp (33.32 to 69.44) and Temp+Temp (65.24 to 85.39).

Taken together, these results support the same conclusion as the image experiments: adaptive compositional adversaries are most useful when failure cases are rare, structured, and temporally or sequentially dependent. The strongest gains consistently appear on the hard combined and temporal video attacks that random augmentation is least likely to cover well.

J Payload Scaling

CAT improves convergence across payload sizes, with especially large gains at low payloads. Payload determines how many bits the model must embed and recover, and thus induces a trade-off between optimization difficulty and attainable robustness. Higher payloads are more difficult to train because the decoder must recover more bits under the same distortion budget, but they also correspond to higher effective capacity and therefore potentially stronger robustness once learned. By contrast, lower payloads are easier to fit under identity-only training, yet this regime is often more brittle because good clean-input performance can be achieved without learning invariance to realistic corruptions. An effective watermarking method should therefore provide both high capacity and robust performance across payload sizes, rather than relying on the deceptively easy low-payload identity regime. Figure 16 makes this distinction clear. CAT improves convergence across the full range from 32 to 256 bits, with especially large gains at lower payloads, where it reduces validation error much more rapidly than random augmentation and in some cases even converges faster than the no-augmentation baseline. At the same time, CAT retains the lowest error on the augmented validation setting, showing that this speedup does not come from overfitting to the identity case.

Random augmentation performs particularly poorly at low payloads because the underlying clean task is already easy: the large gap between easy and hard attacks causes uniformly sampled augmentations to waste many updates on uninformative samples, yielding a brittle model that remains undertrained on the rarer corruptions that actually determine robustness. This effect is less visible in the identity-only baseline because the training objective is then aligned with the easiest validation case, but that apparent simplicity does not translate into robustness under realistic post-processing. CAT avoids this failure mode by adaptively concentrating training pressure on the transformations that remain challenging for the current model.

K Forward and Backward Transfer

Transfer evaluation protocol. We additionally evaluate whether the robustness learned by CAT transfers across attack depths. In the forward transfer setting, we take models trained with single-step

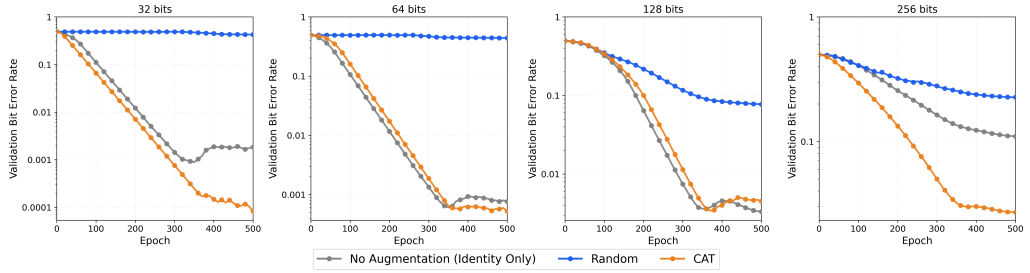


Figure 16: CAT improves PixelSeal training across payload sizes. Validation bit error rate over training is shown for 32-, 64-, 128-, and 256-bit payloads under no augmentation (gray), random augmentation (blue), and CAT (orange). CAT consistently drives the error lower than random augmentation and remains effective as payload increases, whereas random augmentation plateaus at substantially higher error.

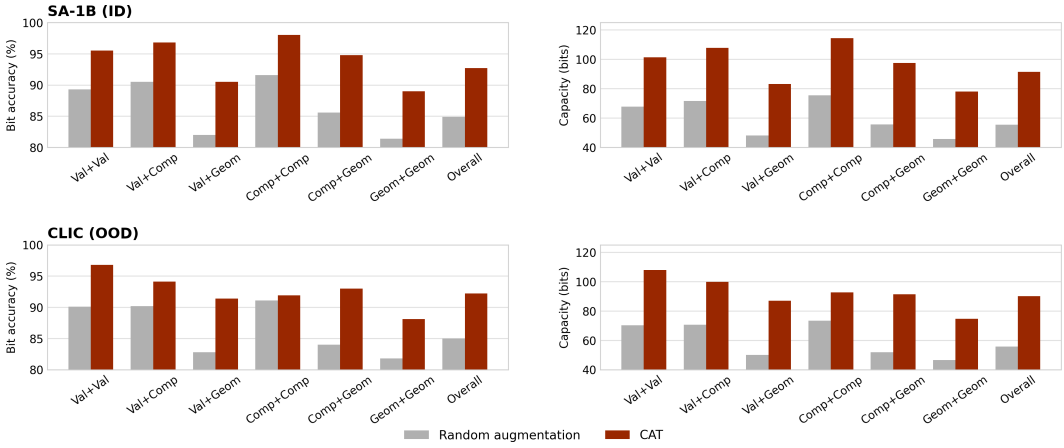


Figure 17: **Forward transfer from single-step training to two-step attack compositions.** Gray bars denote the random-augmentation training baseline and red bars denote CAT. CAT improves both bit accuracy and capacity across all reported composition families on SA-1B and CLIC, with the largest gains on the harder mixed and repeated compositions.

attacks and test them on two-augmentation compositions. In the backward transfer setting, we take models trained with compositional two-step attacks and test them on single augmentations. Figures 17 and 18 summarize PixelSeal results on in-distribution SA-1B and out-of-distribution CLIC using the same bit-accuracy and capacity metrics as the main image experiments.

Forward transfer shows that CAT learns robustness that generalizes to stronger, unseen attack compositions. Figure 17 shows that the single-step CAT model substantially outperforms the single-step random-augmentation baseline when both are evaluated on two-step attack compositions. The gains are broad rather than isolated: on both SA-1B and CLIC, CAT improves both bit accuracy and capacity across every reported composition family, with especially large jumps in the harder mixed and repeated regimes. This pattern suggests that even when CAT is trained against only single-step adversaries, it still learns features that generalize to stronger composed perturbations.

Backward transfer remains strong, indicating limited over-specialization to composed attacks. Figure 18 shows the converse evaluation: models trained on compositional attacks remain competitive when the test-time attack collapses back to a single augmentation. Across both SA-1B and CLIC, compositional CAT improves most single-step families and the overall metrics, while geometric capacity also increases slightly. Overall, the figure shows that compositional CAT learns robustness that transfers back to simpler single-step corruptions.

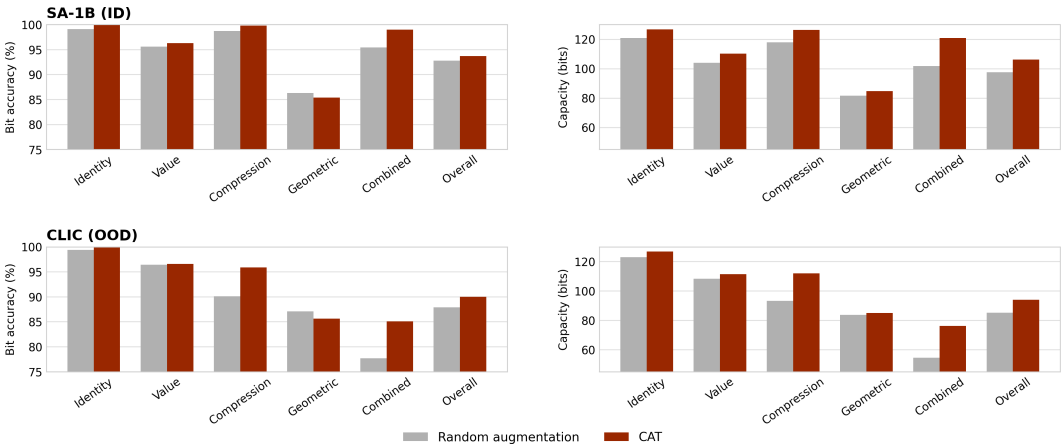


Figure 18: **Backward transfer from compositional training to single-step attacks.** Gray bars denote the random-augmentation training baseline and red bars denote CAT. Compositional CAT improves most single-step families and the overall metrics on both SA-1B and CLIC, while geometric bit accuracy dips slightly even as geometric capacity improves.



Figure 19: Additional qualitative results.

L Qualitative Results and Watermark Residual Analysis

Figures 19–23 present additional qualitative examples for the image watermarking experiments discussed in the main paper.



Figure 20: Additional qualitative results.

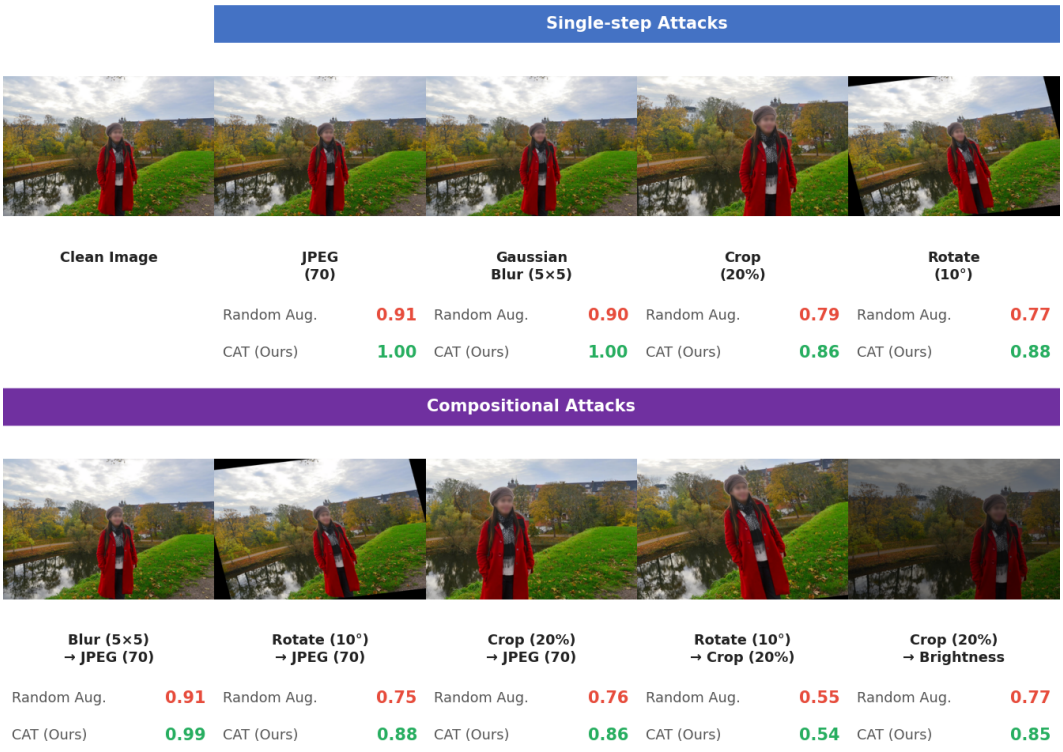


Figure 21: Additional qualitative results.

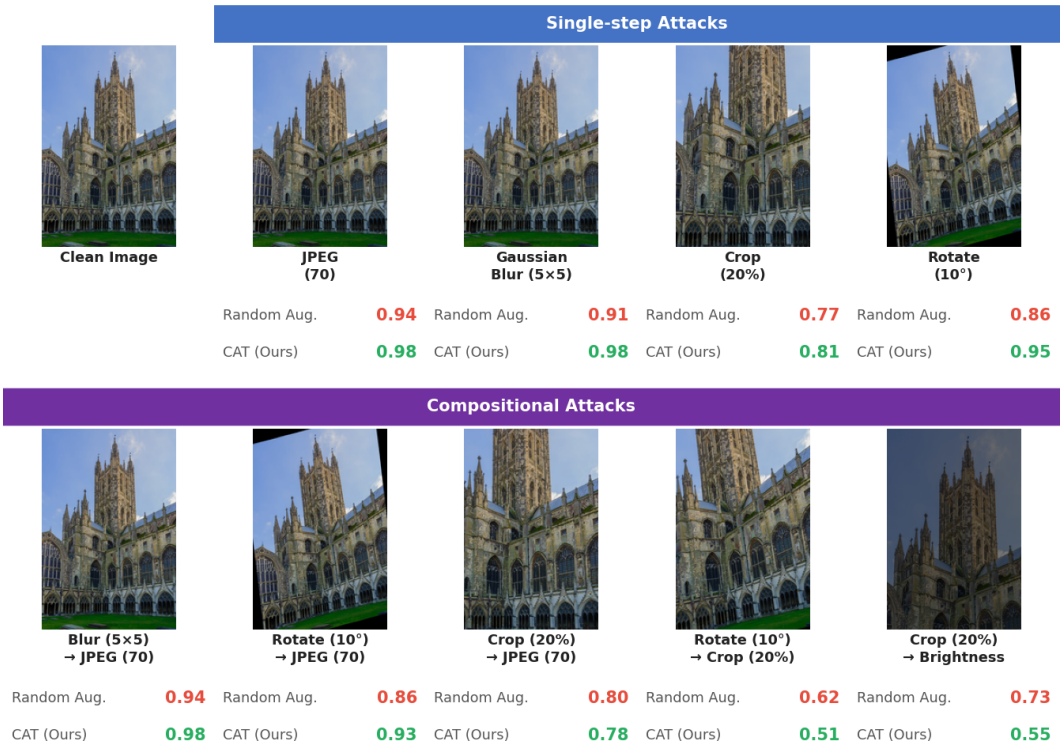


Figure 22: Additional qualitative results.



Figure 23: Additional qualitative results.

L.1 Spatial and Frequency Analysis of Watermark Residuals

Reading the visualizations. Each figure shows mean absolute spatial residuals on the left and mean FFT log-magnitude on the right, with random augmentation above CAT. Low spatial frequencies lie near the center of the Fourier panels. The plots have individual color bars, so compare peak locations and spatial patterns rather than brightness across panels. The Fourier maps visualize spectral structure, not normalized energy fractions; the amplitude comparison comes from the residual measurements, not the displayed log-magnitude scale.

CAT changes residual structure without a comparable increase in amplitude. We compare final-epoch CAT checkpoints with matched random-augmentation baselines for PixelSeal, VideoSeal 0.0, and VideoSeal 1.0 at both training depths. The analysis uses the true pixel residual $r = x_{\mathrm{wm}} - x$ over all test images, before post-processing attacks. Across these six comparisons, changes in mean absolute residual and root-mean-square amplitude remain within one reported standard deviation. The spatial maps are also visually similar. Together with the image-fidelity results in Figure 5, these measurements show that CAT reorganizes the watermark without substantially increasing its amplitude.

Single-step training favors lower spatial frequencies. At $T = 1$, the random-augmentation spectra exhibit diagonally arranged periodic peaks, and CAT shifts the dominant structure toward the frequency origin across all three models (Figures 24, 25, and 26). This spectral shift is consistent with single-step smoothing and compression attacks suppressing fine, high-frequency structure and making lower-frequency watermark components more useful for recovery.

Two-step compositions induce model-dependent spectral changes. At $T = 2$, the radial shift is comparatively small for PixelSeal and VideoSeal 0.0, although lobe shape and orientation change; VideoSeal 1.0 instead shifts its dominant structure outward, reversing the depth-1 trend (Figures 27, 28, and 29). Mixed compositions combine smoothing with geometric or nonlinear effects, producing model- and depth-dependent spectral structures. CAT's frequency redistribution therefore follows the composition and the backbone.

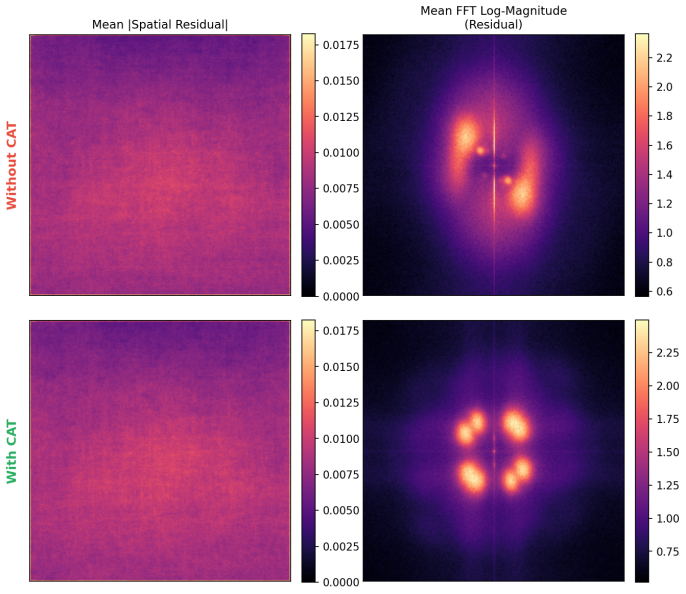


Figure 24: **Single-step CAT training shifts PixelSeal's dominant spectral structure toward lower frequencies.** PixelSeal at $T = 1$. Top: random augmentation; bottom: CAT. Left: mean absolute spatial residual; right: mean FFT log-magnitude of $x_{\mathrm{wm}} - x$. The spatial residual maps remain similar, while the dominant Fourier lobes move inward and become more symmetric. Each panel retains its own color scale.

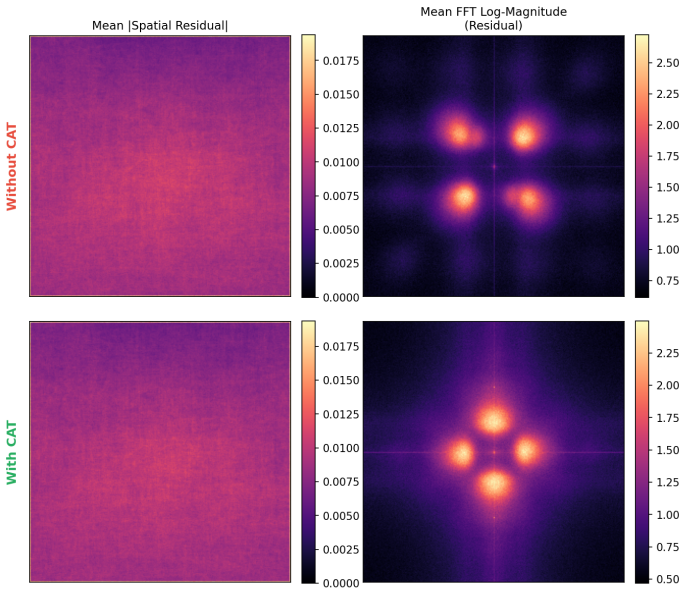


Figure 25: **Single-step CAT training moves VideoSeal 0.0's spectral peaks inward and changes their orientation.** VideoSeal 0.0 at $T = 1$. Top: random augmentation; bottom: CAT. Left: mean absolute spatial residual; right: mean FFT log-magnitude of $x_{\text{wm}} - x$. The diagonal baseline lobes are replaced by a more central, axis-aligned arrangement under CAT, without a comparable change in the spatial residual map. Each panel retains its own color scale.

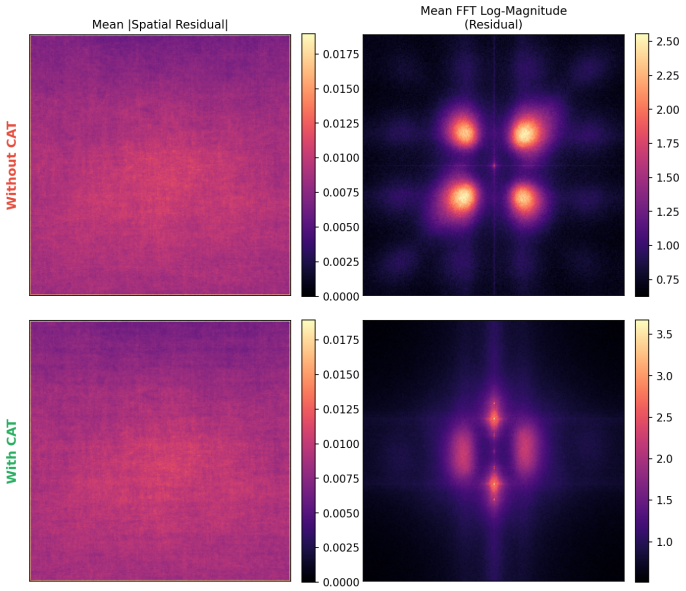


Figure 26: **Single-step CAT training concentrates VideoSeal 1.0's dominant spectral structure nearer the origin.** VideoSeal 1.0 at $T = 1$. Top: random augmentation; bottom: CAT. Left: mean absolute spatial residual; right: mean FFT log-magnitude of $x_{\text{wm}} - x$. The baseline's diagonal lobes give way to more central and axis-aligned components, consistent with the lower-frequency trend at depth 1. Each panel retains its own color scale.

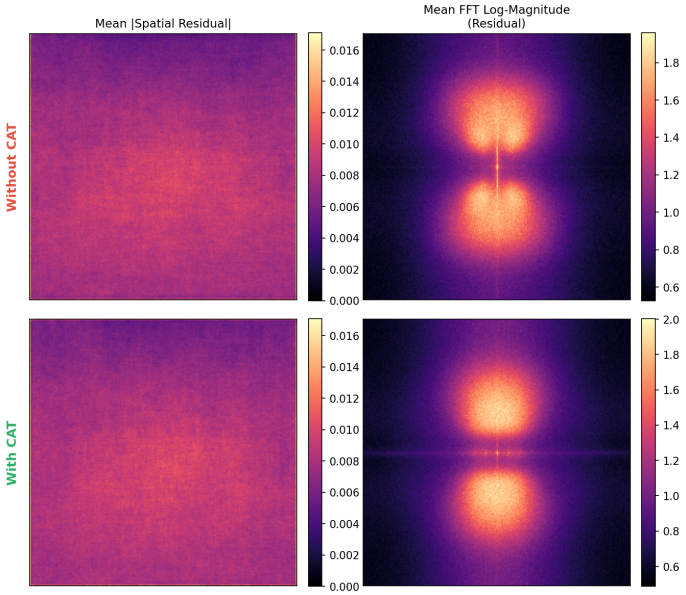


Figure 27: **Two-step CAT training changes PixelSeal's lobe structure with little net radial displacement.** PixelSeal at $T = 2$. Top: random augmentation; bottom: CAT. Left: mean absolute spatial residual; right: mean FFT log-magnitude of $x_{\text{wm}} - x$. The dominant upper and lower spectral regions remain at similar distances from the origin. Each panel retains its own color scale.

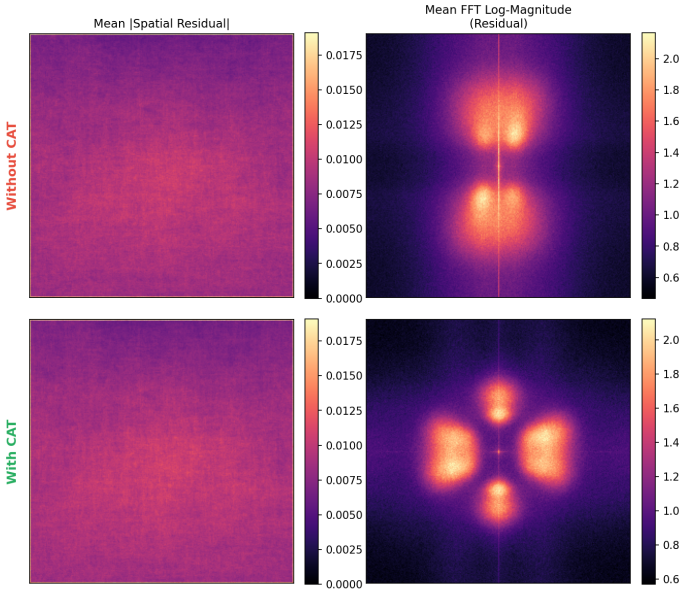


Figure 28: **Two-step CAT training reorganizes VideoSeal 0.0's spectrum without the clear inward shift seen at depth 1.** VideoSeal 0.0 at $T = 2$. Top: random augmentation; bottom: CAT. Left: mean absolute spatial residual; right: mean FFT log-magnitude of $x_{\text{wm}} - x$. CAT changes the orientation and separation of the spectral lobes, illustrating adaptation beyond a simple radial shift to lower frequencies. Each panel retains its own color scale.

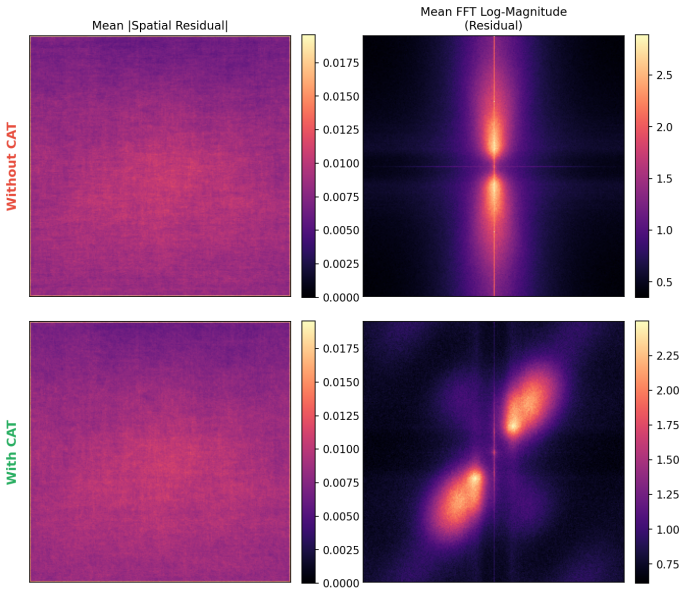


Figure 29: **Two-step CAT training reverses VideoSeal 1.0's lower-frequency trend.** VideoSeal 1.0 at $T = 2$. Top: random augmentation; bottom: CAT. Left: mean absolute spatial residual; right: mean FFT log-magnitude of $x_{\text{wm}} - x$. Relative to the baseline's central vertical structure, CAT produces prominent diagonal lobes farther from the origin, demonstrating model- and depth-dependent spectral redistribution. Each panel retains its own color scale.

M Broader Impact

The work has clear positive applications in provenance tracking, creator attribution, and misuse detection for synthetic or redistributed media. At the same time, stronger watermarking methods can be used in restrictive content-control settings, and stronger attack models can also facilitate watermark-removal research. We believe that exposing brittle robustness claims before deployment is beneficial, but any release of code or trained adversaries should be accompanied by clear documentation of these dual-use risks and by usage restrictions that discourage misuse.