NOT ALL RECOVERED SYMBOLS ARE EQUAL: RELIABILITY-AWARE WATERMARK VERIFICATION FOR AUTOREGRESSIVELY GENERATED IMAGES

Anonymous authors
Paper under double-blind review

ABSTRACT

Recovering verification symbols through retokenization is a key step in verifying watermarks in autoregressively generated images. However, the recovered symbols may differ from those used during generation, especially after image transformations. Importantly, these recovery errors are not uniform: some symbols remain stable, whereas others are more likely to change. Existing autoregressive (AR) image watermarking methods, however, typically treat recovered verification evidence uniformly despite substantial differences in recovery reliability. This limitation may lead to watermark verification failure. To address this issue, we propose Reliability-Aware Verification (RAV), which predicts symbol survival reliability from recovery-side features, including quantization geometry, posterior uncertainty, consistency, and spatial context, and uses the predicted reliability to weight watermark evidence. By emphasizing reliable symbols and suppressing unreliable ones, RAV reduces the risk of verification failure caused by unreliable recovered evidence. Experiments on WMAR and CLUSTERMARK across three autoregressive image generation models, RAR-XL, TAMING, and LlamaGen GPT-B, show that RAV consistently improves watermark verification under diverse transformations at strict false-positive rates.

1 INTRODUCTION

Autoregressive (AR) generation has emerged as a prominent direction for image synthesis, representing images as sequences of discrete visual tokens within sequential generation frameworks (Lu et al., 2024a; Sun et al., 2024; Yu et al., 2025). This tokenized generation paradigm also makes it possible to adapt statistical watermarking techniques developed for language models to visual-token sequences. At verification time, however, the detector does not observe the original visual-token sequence and must recover watermark symbols by retokenizing the rendered image. This recovery is inherently imperfect: decoding visual tokens into pixels and retokenizing the resulting image can alter the original symbols, while common image transformations such as resizing, compression, and screenshotting can further amplify the mismatch. Consequently, watermark verification must operate on imperfectly recovered symbols, making robust detection under symbol mismatch a central challenge for autoregressive image watermarking.

Existing AR image watermarks mainly improve this recovery step: WMAR strengthens reverse cycle-consistency (RCC) and synchronization (Jovanovic et al., 2025), CLUSTERMARK uses visual-token clustering and learned recovery predictors (Lukovnikov et al., 2026), and related strategies further reduce retokenization mismatch (Fernandez et al., 2026; Wu et al., 2025). Yet the final detector still largely treats recovered evidence uniformly, so a stable token or transition can receive essentially the same decision weight as one prone to recovery error. This equal treatment becomes problematic after transformations, when recovery reliability varies substantially across symbols.

A natural way to address this limitation is to perform reliability-aware non-uniform weighting of recovered symbols during watermark detection. However, this approach rests on a key prerequisite: symbol recovery reliability must exhibit some degree of stability and predictability; if reliability differences across symbols were entirely random, they could not support effective weight assignment. To examine this prerequisite, Fig. 1 plots each visual-token position as one point. Its horizontal

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

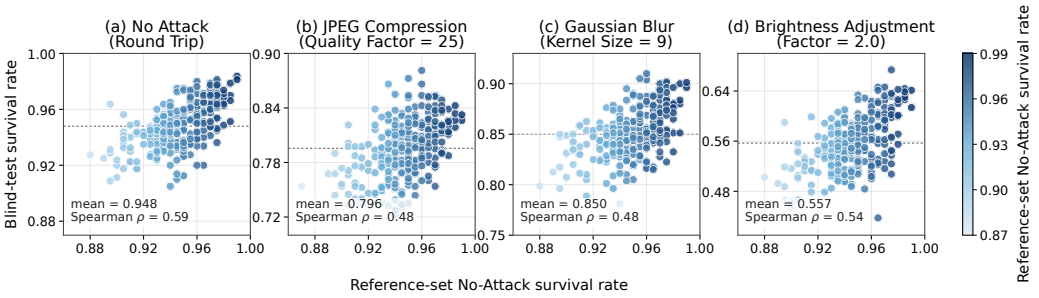

Figure 1: Position-wise consistency of verification-symbol survival in a representative WMAR/RAR-XL setting. Each point denotes one of 256 visual-token positions. The horizontal coordinate and color show No-Attack survival on the 200-image reference set; the vertical coordinate shows survival on the disjoint 800-image blind-test set. Positive Spearman correlations across the four conditions indicate that positions stable on the reference set tend to remain stable after transformations.

coordinate and color indicate the survival rate under the No-Attack condition on a reference set, while its vertical coordinate indicates the survival rate on a disjoint blind-test set under different transformations. Although JPEG compression, Gaussian blur, and brightness adjustment change the overall recovery level, the position-wise survival rates on the reference set remain positively correlated with those on the blind-test set across all conditions: positions that are more stable on the reference set also tend to remain more stable on the blind-test set and after transformations. These results show that survival reliability is structured, supporting reliability-aware weighting.

These findings motivate reliability-aware non-uniform weighting, but realizing this idea raises three challenges. First, recovery-side features beyond visual-token position must predict symbol reliability. Second, these features must be mapped to symbol-level reliability estimates. Third, the estimates must be translated into principled weights for different watermark evidence structures.

To address these challenges, we propose Reliability-Aware Verification (RAV), a detector-side adaptive weighting framework. RAV predicts symbol survival reliability from recovery-side information without access to the secret key, green/red assignment, watermark label, or attack identity, and converts the predictions into statistically motivated, structure-aware weights for keyed evidence. The weighted evidence is aggregated with a variance-normalized statistic calibrated on an independent negative set for strict-FPR decisions. We instantiate RAV for unary WMAR evidence and CLUSTERMARK transitions without modifying generation or embedding.

Our contributions are:

- To our knowledge, we are the first to systematically study *verification-symbol survival reliability* as a detector-side signal for autoregressive image watermark verification, showing that it is a structured and partially predictable property that can support reliability-aware evidence weighting.
- We introduce a unified RAV framework that predicts symbol survival reliability from recovery-side information and converts the predictions into watermark-structure-aware evidence weights. A statistical analysis further shows why evidence with higher survival reliability should receive larger detection weights.
- We demonstrate the effectiveness and generalizability of RAV on two autoregressive image watermarking schemes, WMAR and CLUSTERMARK, across three image generators, RAR-XL, TAMING, and LlamaGen GPT-B, with consistent gains over the corresponding original detectors under blind testing, diverse transformations, unseen neural compression, and regeneration attacks, particularly at strict FPRs.

2 RELATED WORK

Generative-image watermarking and robustness. An important line of generative-image watermarking integrates watermarks directly into generation, with diffusion models as a major technical

substrate. Representative methods include Stable Signature, Tree-Ring, Gaussian Shading, RingID, DiffuseTrace, WMAdapter, and SAT-LDM (Fernandez et al., 2023; Wen et al., 2023; Yang et al., 2024; Ci et al., 2024a; Lei et al., 2024; Ci et al., 2025; Zhang and Zeng, 2025). Related work has expanded to localized watermarking and content attribution (WAM and ProMark), dynamic marks, tamper/semantic awareness, and resistance to model fine-tuning (Sander et al., 2025; Asnani et al., 2024; Chen et al., 2025a;b; Arabi et al., 2025; Wang et al., 2025). Robustness under compression, editing, and regeneration remains challenging: benchmarks and removal, transfer, paraphrasing, regeneration, and forgery studies show that perturbations can weaken, remove, or forge watermark evidence (An et al., 2024; Lu et al., 2025; Zhao et al., 2024; Liu et al., 2025; Muller et al., 2025; Hu et al., 2025; Dixit et al., 2026).

Autoregressive image generation and watermarks. Discrete visual generation has evolved from vector quantization (VQ) transformers and masked-token models to large-scale autoregressive systems, including next-scale, continuous-valued, and hierarchical generation (Esser et al., 2021; Chang et al., 2022; Yu et al., 2022; Sun et al., 2024; Yu et al., 2025; Tian et al., 2024; Li et al., 2024). Visual-token generation also enables generation-time watermarking and transfers red–green statistics from language models (Kirchenbauer et al., 2023; 2024). WMAR improves exact-token recovery via RCC and synchronization, while SyncSeal strengthens geometric synchronization (Jovanovic et al., 2025; Fernandez et al., 2026); ClusterMark and C-reweight reduce retokenization mismatch through clustering (Lukovnikov et al., 2026; Wu et al., 2025); and IndexMark and BitMark explore alternative AR mechanisms (Tong et al., 2025; Kerner et al., 2025). VQAGI-Mark embeds multi-bit watermarks through visually similar codebook-index replacements (Lai and Bors, 2026), while GenPTW supports latent-space watermarking for both visual AR and diffusion models with provenance tracing and tamper localization (Gan et al., 2026). Recent work also exposes removal and forgery vulnerabilities in AR watermarks (Muller et al., 2026). Overall, prior methods focus mainly on preservation or recovery, leaving reliability differences among recovered symbols largely unmodeled; RAV instead exploits this recovery reliability during detection.

Reliability-aware statistical detection. Prior text-watermarking studies have recognized that tokens can contribute unequally to watermark detection. SWEET (Lee et al., 2024) uses token entropy to select high-entropy positions for watermarking and detection, while EWD (Lu et al., 2024b) further assigns entropy-dependent weights to token-level evidence. Adaptive watermarking methods (Liu and Bu, 2024) similarly exploit generation-time uncertainty to adjust watermarking decisions. These studies demonstrate the value of non-uniform token-level weighting in text watermarking, where the observed text can typically be retokenized directly into discrete language tokens without passing through a lossy continuous reconstruction process. Autoregressive image watermarking introduces an additional challenge. The original visual tokens are first decoded into pixels, and verification must later recover discrete visual tokens from the resulting image through re-encoding and quantization. This recovery process is inherently lossy and can be further disrupted by image transformations, making the reliability of recovered watermark evidence vary substantially across symbols. RAV therefore focuses on this post-recovery uncertainty. It predicts the survival reliability of recovered visual tokens or clusters and converts the predicted reliability into evidence weights according to the structure of the underlying watermark detector.

3 Reliability-Aware Verification

RAV comprises a unified reliability-aware verification framework and two watermark-specific instantiations. Figure 2 organizes it into three stages: (i) unchanged generation and recovery, providing original/recovered verification symbols and recovery-side information; (ii) offline key-independent reliability training, which constructs symbol-survival labels and learns a frozen predictor; and (iii) online reliability-aware verification, which combines predicted reliabilities with keyed watermark evidence to form a normalized detection statistic. Sections 3.2 and 3.3 instantiate these components for WMAR and ClusterMark, respectively, as summarized in Table 1.

3.1 Unified RAV Framework

Generation and recovery. Let the original verification-symbol sequence be $S = (s_1, \ldots, s_T)$, where $t \in \{1, \ldots, T\}$ indexes a position on the token grid. RAV does not alter the generator,

decoder, watermark key, or embedding rule. After image decoding and a possible transformation, the frozen recovery module produces the recovered sequence $\hat{S} = (\hat{s}_1, \ldots, \hat{s}_T)$ together with key-independent recovery-side features $\Phi = (\phi_1, \ldots, \phi_T)$. The same recovery interface is used during offline reliability training and online verification.

Offline reliability training. During training, the original symbol sequence is available, so each recovered symbol can be assigned a survival label

$$y_t = \mathbf{1}[\hat{s}_t = s_t]. \tag{1}$$

The reliability predictor f_θ maps the complete recovery feature field to an occurrence-specific survival probability,

$$r_t = \sigma(f_\theta(\Phi)_t) \approx \Pr(y_t = 1 \mid \Phi), \qquad r_t \in [0, 1], \tag{2}$$

where r_t estimates the probability that the recovered symbol at position t matches the original verification symbol and σ is the sigmoid function. A spatial predictor may use neighboring features $\phi_{t'}$, but r_t remains specific to the current image and symbol occurrence rather than being a fixed positional prior. We use binary cross-entropy as the training loss,

$$\mathcal{L}_{\text{surv}} = -\frac{1}{|\mathcal{D}|} \sum_{(i,t)\in\mathcal{D}} \Big[y_{it} \log r_{it} + (1 - y_{it}) \log(1 - r_{it}) \Big]. \tag{3}$$

After minimizing $\mathcal{L}_{\text{surv}}$, f_θ is frozen for verification; Sec. 4 and Appendix A specify the split and leakage-control rules.

Online reliability-aware verification. For a suspect image, the frozen recovery module and reliability predictor first produce $\hat{S}$, Φ, and the reliability map r. The original keyed detector operates in parallel. We use m to denote a generic *evidence item*: for unary WMAR evidence, $m = t$; for ClusterMark, $m = t$ denotes the cluster transition ending at position t. Let $g_m(k) \in \{0, 1\}$ indicate whether the recovered evidence item is a green event under secret key k, and let γ denote the corresponding green probability under the null hypothesis. A watermark-specific composition function $w_m = \Psi_m(r)$ converts symbol reliability into evidence reliability. The final normalized statistic is

$$Z_{\text{RAV}} = \frac{\sum_{m\in\mathcal{I}} w_m \left[g_m(k) - \gamma \right]}{\sqrt{\gamma(1-\gamma) \sum_{m\in\mathcal{I}} w_m^2}}, \tag{4}$$

where $\mathcal{I}$ is the set of valid, context-complete, duplicate-filtered evidence items. The secret key enters only through $g_m(k)$; both r_t and w_m are computed without access to the key. The operational threshold τ is calibrated only on an independent negative calibration set and then frozen; the final decision is $\mathbf{1}[Z_{\text{RAV}} > \tau]$. Because AR contexts, repeated n-grams, and cluster transitions induce dependence, we do not interpret Z_{RAV} as an exact finite-sample Gaussian p-value.

Statistical rationale for reliability weighting. A simplified survival-corruption model explains why survival probability is a natural linear weight. Suppose a surviving evidence item contributes a mean watermark signal proportional to its survival probability ρ_m, whereas a corrupted item contributes approximately zero-mean noise with comparable conditional variance. For a linear weighted statistic, the resulting signal-to-noise ratio is proportional to

$$\frac{\sum_m w_m \rho_m}{\sqrt{\sum_m w_m^2}} \le \sqrt{\sum_m \rho_m^2}, \tag{5}$$

where the inequality follows from the Cauchy–Schwarz inequality, with equality when $w_m \propto \rho_m$. Thus, under this simplified survival-corruption model, reliability-proportional weighting is SNR-optimal among linear weighting rules. This result motivates assigning larger weights to evidence with higher predicted survival reliability, rather than requiring the practical weighting function to exactly equal the estimated survival probability. For unary WMAR evidence, the idealized reliability is $\rho_t = r_t$, while for a ClusterMark transition we use $\rho_t \approx r_{t-1} r_t$ under a conditional endpoint-independence approximation. In practice, RAV-WMAR applies a small amount of shrinkage toward equal weighting, as specified in Sec. 3.2. This argument provides a statistical motivation rather than an exact model of AR dependence; the empirical negative calibration of τ absorbs the dependencies present in the operational detector.

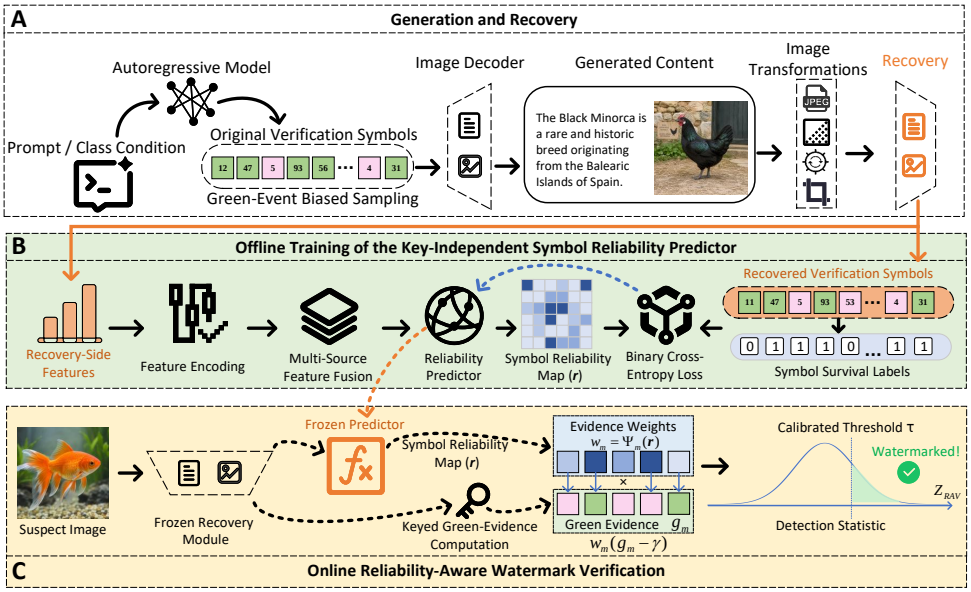

Figure 2: Unified RAV framework. (A) Generation and recovery remain unchanged. (B) A key-independent predictor learns symbol survival reliability offline. (C) Predicted reliabilities are converted into structure-aware evidence weights for calibrated watermark verification.

Table 1: The two RAV instantiations. Neither reliability predictor receives the secret key or green/red assignment.

Instance	Verification symbol	Reliability features/model	Evidence weight
RAV-WMAR	VQ token	Top-8 codebook distances, quantization gap, soft assignment, latent/consistency cues; residual spatial CNN (231,553 params.)	$w_t = \epsilon + (1-\epsilon)r_t$ $\epsilon = 0.05$
RAV-ClusterMark	Visual cluster	64-class posterior with uncertainty/consistency channels; unary spatial CNN (39,585 params.)	$w_t = r_{t-1}r_t$

3.2 RAV-WMAR: Unary Token Reliability

For WMAR, the verification symbol is the exact VQ token, i.e., $s_t = q_t$ and $\hat{s}_t = \hat{q}_t$. The detector first constructs a 16×16 feature field Φ^{W}. At each position, ϕ_t^{W} concatenates the Top-8 codebook-distance vector, the relative quantization gap and soft-assignment statistics (including entropy), a fixed projection of the continuous pre-quantization latent, multi-view recovery-consistency cues, spatial coordinates, a token-ID embedding, and the handcrafted quantization-margin prior. Let $F_{\theta_{\mathrm{W}}}$ denote the frozen residual spatial CNN. Equation (2) then becomes

$$r_t^{\mathrm{W}} = \sigma\big(F_{\theta_{\mathrm{W}}}(\Phi^{\mathrm{W}})_t\big) \approx \Pr(\hat{q}_t = q_t \mid \Phi^{\mathrm{W}}). \tag{6}$$

The training target $y_t^{\mathrm{W}} = \mathbf{1}[\hat{q}_t = q_t]$ is independent of the secret key. At detection time, the secret key and recovered AR context h_t jointly define the WMAR green list $G_k(h_t)$, yielding

$$g_t^{\mathrm{W}}(k) = \mathbf{1}[\hat{q}_t \in G_k(h_t)], \qquad w_t^{\mathrm{W}} = \epsilon + (1-\epsilon)r_t^{\mathrm{W}}, \quad \epsilon = 0.05. \tag{7}$$

As summarized in Figure 3, the generic evidence index in Eq. (4) is $m = t$, yielding weighted evidence $w_t^{\mathrm{W}}[g_t^{\mathrm{W}}(k) - \gamma]$ for each recovered token. The small constant $\epsilon = 0.05$ provides conservative shrinkage toward equal weighting, reducing the influence of potentially overconfident low-reliability predictions while preserving the reliability-dependent ordering of the evidence weights. This value is fixed before blind evaluation.

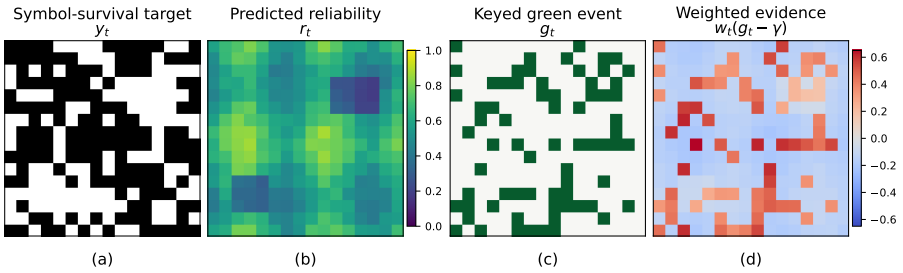

Figure 3: RAV-WMAR evidence construction: predicted token survival reliability weights the keyed green-token evidence to form the final reliability-weighted evidence in Eq. (4).

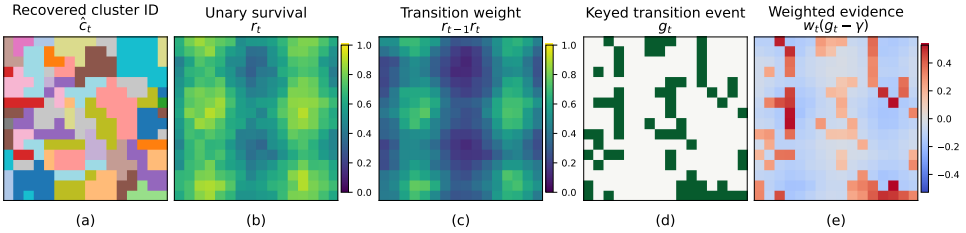

Figure 4: RAV-ClusterMark evidence construction: cluster survival reliabilities are combined into transition weights, which weight keyed transition events to form the final evidence in Eq. (4).

3.3 RAV-ClusterMark: Transition Reliability

For ClusterMark, the verification symbol is the cluster identity of the generated token, $s_t = c(q_t)$. The official Cluster Predictor outputs a 64-class posterior π_t for the recovered cluster. To capture recovery reliability beyond the posterior distribution alone, we augment this output with five additional uncertainty and consistency features derived from the recovery process. We then reshape these features together with the posterior on the 16×16 grid, obtaining a 69-channel recovery field Φ^{C}. The main unary spatial CNN $F_{\theta_{\mathrm{C}}}$ predicts the survival probability of each endpoint,

$$r_t^{\mathrm{C}} = \sigma\big(F_{\theta_{\mathrm{C}}}(\Phi^{\mathrm{C}})_t\big) \approx \Pr(\hat{c}_t = c_t \mid \Phi^{\mathrm{C}}). \tag{8}$$

ClusterMark watermark evidence is transition-based. Let $\mathcal{G}_k^{\mathrm{C}}$ denote the secret-key-defined set of green cluster transitions. The evidence item ending at position t is

$$g_t^{\mathrm{C}}(k) = \mathbf{1}[(\hat{c}_{t-1}, \hat{c}_t) \in \mathcal{G}_k^{\mathrm{C}}], \qquad w_t^{\mathrm{C}} = r_{t-1}^{\mathrm{C}} r_t^{\mathrm{C}}. \tag{9}$$

Figure 4 illustrates the RAV-ClusterMark construction: the spatial CNN predicts endpoint survival reliabilities r_t^{C}, whose product forms the transition weight $w_t^{\mathrm{C}} = r_{t-1}^{\mathrm{C}} r_t^{\mathrm{C}}$ and yields weighted evidence $w_t^{\mathrm{C}}[g_t^{\mathrm{C}}(k) - \gamma]$. For ablation, we retain a pointwise MLP, confidence baselines, and a pairwise CNN that directly predicts transition survival. For LlamaGen GPT-B, a frozen generator-specific adapter maps VQ prequant features to the same 64-class posterior interface ($T = 0.75$, $\beta = 2$); the subsequent RAV equations remain unchanged.

4 Experimental Setup

Evaluation principles. Survival predictability is evaluated by AUROC for both WMAR and ClusterMark. For watermark verification, following the respective base-method protocols, WMAR reports TPR at 1% and 0.1% FPR (Jovanovic et al., 2025), with both independently calibrated thresholds frozen before blind evaluation. ClusterMark reports AUROC and TPR@1% FPR (Lukovnikov et al., 2026); we further conduct a prefix-disjoint blind test, freezing an independently calibrated 1% operating point and reporting 0.1% TPR as a held-out ROC readout. For both watermark families, confidence intervals use image-paired bootstrap; transformed copies stay grouped, and blind results never guide model selection. Appendix A gives calibration and implementation details.

Table 2: WMAR verification at strict FPRs under clean (None), value transformations (Val.), geometric (Geo.), adversarial purification (Adv.), and unseen neural compression (NC). Overall excludes NC; the last column reports realized wrong-key FPR on the blind split.

Operating point	Generator	Detector	None	Val.	Geo.	Adv.	NC	Overall	Δ vs. original	Wrong-key FPR
TPR@1% FPR	TAMING (256×256)	Original WMAR detector	.9925	.8391	.8021	.6988	.7927	.8282	–	.00913
		Handcrafted reliability margin	.9925	.8556	.8167	.7413	.8250	.8451	+.0169	.01026
		RAV-WMAR	**.9950**	**.8663**	**.8313**	**.7900**	**.8673**	**.8604**	**+.0322**	.01039
		Permuted RAV-WMAR	.9925	.8109	.7567	.6113	.7417	.7908	-.0374	.01053
	RAR-XL (256×256)	Original WMAR detector	.9963	.8913	.8338	.7975	.9488	.8733	–	.01024
		Handcrafted reliability margin	.9963	.9047	.8388	.8388	.9648	.8856	+.0122	.01012
		RAV-WMAR	**.9975**	**.9128**	**.8458**	**.8688**	**.9706**	**.8950**	**+.0217**	.01033
		Permuted RAV-WMAR	.9950	.8684	.7988	.7150	.9133	.8422	-.0311	.01048
TPR@0.1% FPR	TAMING (256×256)	Original WMAR detector	.9925	.8006	.7642	.5763	.7083	.7849	–	.00090
		Handcrafted reliability margin	.9925	.8213	.7875	.6250	.7465	.8072	+.0224	.00100
		RAV-WMAR	**.9938**	**.8378**	**.8113**	**.6950**	**.7908**	**.8304**	**+.0456**	.00089
		Permuted RAV-WMAR	.9888	.7556	.6888	.4875	.6323	.7294	-.0554	.00093
	RAR-XL (256×256)	Original WMAR detector	.9938	.8616	.8163	.6713	.9094	.8400	–	.00097
		Handcrafted reliability margin	.9950	.8806	.8271	.7375	.9350	.8596	+.0196	.00121
		RAV-WMAR	**.9963**	**.8922**	**.8338**	**.7700**	**.9448**	**.8707**	**+.0307**	.00104
		Permuted RAV-WMAR	.9938	.8344	.7304	.5625	.8513	.7872	-.0528	.00104

Baselines and controls. For WMAR, the original equal-weight detector is primary, with handcrafted quantization-margin weighting, RAV-WMAR, and a spatially permuted RAV map as additional comparisons. For ClusterMark, we retain K64 hard VQ, Token Predictor, and hard Cluster Predictor; posterior soft/max probability are recovery-confidence comparators, pointwise/pairwise/unary networks are learned RAV estimators, and permuted unary is the alignment control. Unary CNN is selected on validation and frozen as the main RAV-ClusterMark instance on RAR-XL.

Generator-specific protocols. RAV-WMAR is evaluated on RAR-XL and TAMING. RAR-XL uses the official MaskGIT-VQGAN adaptation/synchronization pipeline and a one-time 200-calibration/800-blind new-image/new-key split over nine canonical conditions, followed by genuine negatives, 28 attack settings, and six unseen CompressAI quality-3 codecs. TAMING uses the RAV architecture/statistic on Taming Transformer + ImageNet VQGAN, with independent 1,000-image grouped out-of-fold (OOF) development and 200/800 blind sets. RAV-ClusterMark uses $K = 64$, $\delta = 5$, $\gamma = 0.25$: RAR-XL uses 2,000 watermarked and 2,000 clean images with fixed 800/400/800 image-disjoint train/validation/test IDs, plus an independent prefix-7 200-calibration/800-blind protocol; LlamaGen GPT-B uses 2,000 clean and 2,000 K64 images with identical splits, followed by eight pixel attacks and three regeneration attacks under the frozen detector.

5 RESULTS

5.1 VERIFICATION-SYMBOL SURVIVAL IS PREDICTABLE

The first premise of RAV is that recovery failures are not random with respect to the information available at verification time. Figure 1 first provides a position-wise diagnostic in the representative WMAR/RAR-XL blind protocol, showing that relative positional stability transfers from the reference set to a disjoint blind-test set under all four conditions. We then directly predict the survival labels of Eq. (1) using only key-independent recovery features, before evaluating any keyed watermark statistic. Across the four watermark/generator instances, survival AUROC reaches 0.9060 for WMAR/RAR-XL, 0.8353 for WMAR/TAMING, 0.8424 for CLUSTERMARK/RAR-XL, and 0.8303 for CLUSTERMARK/GPT-B, all well above the chance level of 0.5. For RAR-XL CLUSTERMARK, supplementary probability-quality and cross-attack diagnostics are provided in Appendix A.1. These results directly support the first hypothesis: verification-symbol survival is predictable from key-independent recovery information before the secret key is introduced.

5.2 RAV-WMAR ROBUSTNESS VERIFICATION

Table 2 compares frozen RAV-WMAR with the original equal-weight detector, handcrafted margin weighting, and a spatial-permutation control.

Table 3: ClusterMark robustness across generators. Cells report held-out AUROC–TPR@1% FPR. Color Jitter and Regeneration are macro averages; bold marks the best non-control value for each metric under each condition.

Model / Detector	Clean	JPEG 20	Blur $R = 3$	Gauss. $\sigma = .2$	S&P .1	Color Jitter	Regeneration
LlamaGen GPT-B 256×256							
Original ClusterMark detectors							
K64 hard VQ (training-free)	**.9999–.9988**	.9791–.8812	.8975–.1550	.8445–.3438	.8339–.2938	.9271–.7178	.9914–.9442
+ Token Predictor	.9988–.9962	.9811–.8725	.8752–**.1963**	.8359–.2938	.8275–.2737	.9226–.7066	.9876–.9100
+ Cluster Predictor, hard	.9965–.9925	.9692–.8187	.8891–.0387	.8332–.3100	.8268–.2700	.9147–.6556	.9897–.9192
Simple recovery confidence comparators							
Posterior soft	.9989–.9962	.9834–.8950	**.9050**–.1537	**.8592**–.3750	**.8521**–.3450	**.9292**–.7466	**.9923**–.9571
Max probability	.9986–.9962	.9830–.8938	.8978–.1175	.8536–.3663	.8478–.3488	.9270–.7469	.9919–.9571
*Learned **RAV** estimators*							
RAV, pointwise MLP	.9985–.9962	**.9846**–.9050	.9000–.1087	.8553–.3600	.8508–.3362	.9282–**.7572**	.9922–**.9592**
RAV, pairwise spatial CNN	.9981–.9962	.9833–**.9087**	.8960–.1087	.8541–.3513	.8512–.3525	.9262–.7497	.9920–**.9592**
RAV-ClusterMark, unary CNN (main)	.9982–.9962	.9842–.9050	.8978–.1212	.8569–**.3762**	.8509–**.3575**	.9270–.7512	.9920–.9550
Alignment control							
Permuted unary control	.9982–.9962	.9840–.9087	.8984–.1212	.8561–.3575	.8505–.3513	.9273–.7588	.9920–.9592
RAR-XL 256×256							
Original ClusterMark detectors							
K64 hard VQ (training-free)	1.0000–1.0000	.9804–.8287	.8343–.1750	.6918–.0688	.7383–.0838	.9190–.6491	.9834–.8550
+ Token Predictor	1.0000–1.0000	.9852–.8775	.9825–.8350	.9580–.7500	1.0000–1.0000	.9903–.9481	.9924–.9050
+ Cluster Predictor, hard	1.0000–1.0000	.9857–.9062	.9774–.7738	.9309–.6625	1.0000–1.0000	.9840–.9344	.9904–.9175
Simple recovery confidence comparators							
Posterior soft	1.0000–1.0000	.9892–.9237	**.9863**–.8600	.9492–.7438	1.0000–1.0000	.9908–.9413	.9936–.9396
Max probability	1.0000–1.0000	.9903–.9363	.9845–.8588	.9590–.8013	1.0000–1.0000	.9932–.9656	.9937–.9392
*Learned **RAV** estimators*							
RAV, pointwise MLP	1.0000–1.0000	.9916–.9413	.9854–.8675	.9631–.8050	1.0000–1.0000	.9930–.9684	.9949–.9533
RAV, pairwise spatial CNN	1.0000–1.0000	.9925–**.9450**	.9852–.8788	.9685–.8087	1.0000–1.0000	.9926–.9688	.9955–**.9583**
RAV-ClusterMark, unary CNN (main)	1.0000–1.0000	**.9931–.9450**	.9856–**.8862**	**.9698–.8150**	1.0000–1.0000	**.9933–.9716**	**.9957**–.9567
Alignment control							
Permuted unary control	1.0000–1.0000	.9738–.8250	.9648–.7275	.9036–.5975	1.0000–1.0000	.9785–.9069	.9851–.8800

On RAR-XL, RAV-WMAR improves equal weighting by 2.17/3.07 points and handcrafted margin by 0.94/1.11 points at 1%/0.1% FPR, with positive 95% bootstrap CIs. TAMING gains are +3.22/+4.56 and +1.53/+2.32 points, respectively, with non-negative gains across all nine conditions. Larger gains at 0.1% FPR indicate greater benefit under stricter false-positive constraints, while permutation confirms the importance of reliability-evidence alignment. Genuine negatives yield FPRs of 0.01048/0.00108 on RAR-XL and 0.01075/0.00099 on TAMING; RAV adds 18.95 ms/image. Full stress-test and codec details are in the appendix.

5.3 RAV-ClusterMark Robustness Verification

Table 3 evaluates RAV on ClusterMark transition evidence against original detectors, simple recovery-confidence comparators, and learned estimators.

On RAR-XL, the validation-selected unary RAV obtains the best six-attack macro AUROC–TPR@1% FPR (0.9896–0.9291), compared with 0.9781–0.8657 for the hard Cluster Predictor. This corresponds to gains of 1.15 and 6.34 percentage points in AUROC and TPR, respectively. Spatial permutation reduces the pair to 0.9676–0.8228, a TPR drop of 10.63 percentage points relative to the aligned unary RAV. These results show that learned, correctly aligned cluster-survival reliability can substantially improve transition verification on RAR-XL. GPT-B shows a different pattern. Posterior soft achieves the highest six-attack macro AUROC and a slightly higher TPR, at 0.9202 and 0.5787; the unary CNN reaches 0.5777 TPR and is nearly tied with the spatially permuted result of 0.5761. This indicates that, on GPT-B, the cluster posterior already carries most of the recovery confidence useful for verification, while learned spatial modeling provides little additional benefit. The conclusion is therefore framework level rather than tied to a particular spatial CNN: the most effective reliability representation and the marginal value of spatial context can depend on the generator/tokenizer. Permutation destroys local alignment but does not erase the global recovery confidence carried by the learned weight distribution. This helps explain why the permuted control can still outperform the original hard detector on GPT-B.

Additional new-prefix blind test. The matched experiments use image-disjoint 800/400/800 train/validation/test splits but share the development hash-prefix condition. We therefore add a stricter image- and prefix-disjoint RAR-XL blind test: images are regenerated under prefix 7, thresholds are calibrated on negative IDs 0–199, and IDs 200–999 are read once for evaluation.

Table 4: Independent RAR-XL prefix-7 blind evaluation. "Emp." uses held-out ROC readouts; "Frozen" applies the pre-blind calibration threshold.

Detector	AUROC	Emp. TPR@1% FPR	Emp. TPR@0.1% FPR	Frozen TPR@1% FPR	Realized FPR@1%
Original ClusterMark detectors					
K64 + Cluster Predictor, hard	.9809	.9096	.8317	.8888	.0061
Simple recovery confidence comparators					
Posterior soft	.9872	.9275	.8758	.9069	.0074
Max probability	.9901	.9465	.9054	.9385	.0079
*Learned **RAV** estimators*					
RAV, pointwise MLP	.9904	.9488	.9149	.9406	.0068
RAV, pairwise spatial CNN	.9914	.9522	.8838	**.9464**	.0076
RAV-ClusterMark, unary CNN (main)	**.9922**	**.9538**	**.9231**	.9440	.0057
Alignment control					
Permuted unary control	.9738	.8613	.7899	.8436	.0074

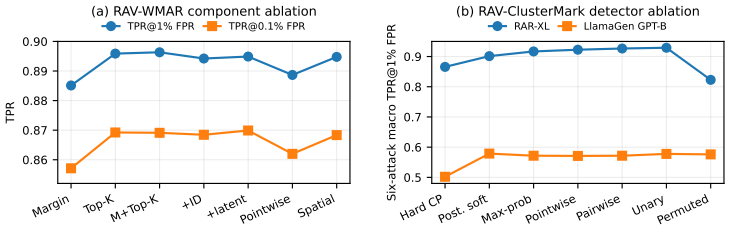

Figure 5: RAV-WMAR (RAR-XL) and RAV-ClusterMark (RAR-XL/GPT-B) ablations.

Under prefix 7, unary RAV reaches 0.9922 AUROC and 0.9538/0.9231 TPR. Its gain over Posterior soft shows that learned survival reliability generalizes beyond the development prefix and adds information beyond raw posterior confidence. At the frozen 1% threshold, its TPR/FPR are 0.9440/0.0057 versus 0.8888/0.0061 for the hard Cluster Predictor. Pairwise CNN is slightly higher in frozen TPR (0.9464), but the unary model remains frozen; permutation sharply degrades detection. Overall, reliability-aware verification provides a framework-level benefit, while the best estimator depends on the generator/tokenizer. Full unary RAV costs 8.534 vs. 3.945 ms/image for hard verification; paired-bootstrap CIs are in Appendix C.1.

5.4 Ablation and Mechanism Analysis

Figure 5 summarizes the frozen ablations. For RAV-WMAR, Top-K quantization geometry provides most of the gain (0.8959/0.8692 vs. 0.8851/0.8571 for margin), while the full spatial model (0.8948/0.8683) outperforms its pointwise counterpart (0.8887/0.8620). For ClusterMark/RAR-XL, posterior soft raises hard-CP TPR from 0.8657 to 0.9014, learned RAV reaches 0.9226–0.9291, and permutation drops unary to 0.8228. Overall, gains mainly come from informative reliability cues and their correct evidence alignment, with spatial modeling providing model-dependent benefits.

6 Discussion and Conclusion

We identify verification-symbol survival reliability as a stable, partially predictable property of recovered AR-image watermark evidence and introduce RAV to convert key-independent survival estimates into structure-aware verification weights. Results on WMAR and ClusterMark across RAR-XL, TAMING, and LlamaGen GPT-B support reliability-aware evidence aggregation as a detector-side principle without changing generation or embedding.

Limitations. RAV cannot recover evidence destroyed by severe desynchronization, so it complements rather than replaces synchronization and robust recovery. It also adds detector-side computation (18.95 ms/image beyond WMAR recovery; 8.534 vs. 3.945 ms/image for ClusterMark). Our null audits use same-generator unwatermarked outputs, leaving cross-source FPR transfer and adaptive attacks open. No estimator is uniformly best: on GPT-B, simple posterior confidence can match learned spatial variants, motivating adaptive reliability estimation.

REPRODUCIBILITY STATEMENT

Section 4 and Appendix Table 5 specify image-disjoint splits, negative-only calibration, model-selection freezes, attack settings, and one-time blind protocols. The supplementary package includes anonymized source code and configurations, machine-readable result tables, experiment records, hashes, and scripts used to reconstruct the reported results. Generator and embedding pipelines remain those of the corresponding WMAR or ClusterMark instance; detector-specific choices are frozen before blind evaluation.

AI USE STATEMENT

A large language model (ChatGPT) was used as a general-purpose assistive tool for language editing, drafting assistance, and programming/documentation support. The authors remain fully responsible for the paper and independently verified the experimental protocols, code executions, numerical results, citations, and scientific claims.

REFERENCES

Pierre Fernandez, Guillaume Couairon, Herve Jegou, Matthijs Douze, and Teddy Furon. The Stable Signature: Rooting watermarks in latent diffusion models. In *IEEE/CVF International Conference on Computer Vision*, 2023.

Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. Tree-Rings Watermarks: Invisible Fingerprints for Diffusion Images. In *Advances in Neural Information Processing Systems*, 2023.

Zijin Yang, Kai Zeng, Kejiang Chen, Han Fang, Weiming Zhang, and Nenghai Yu. Gaussian Shading: Provable performance-lossless image watermarking for diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

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*, 2024.

Jiasen Lu, Christopher Clark, Sangho Lee, Zichen Zhang, Savya Khosla, Ryan Marten, Derek Hoiem, and Aniruddha Kembhavi. Unified-IO 2: Scaling Autoregressive Multimodal Models with Vision Language Audio and Action. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. MaskGIT: Masked generative image transformer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. In *Transactions on Machine Learning Research*, 2022.

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. arXiv:2406.06525, 2024.

Qihang Yu, Ju He, Xueqing Deng, Xiaohui Shen, and Liang-Chieh Chen. Randomized autoregressive visual generation. In *IEEE/CVF International Conference on Computer Vision*, 2025.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In *International Conference on Machine Learning*, 2023.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Manli Shu, Khalid Saifullah, Kezhi Kong, Kasun Fernando, Aniruddha Saha, Micah Goldblum, and Tom Goldstein. On the reliability of watermarks for large language models. In *International Conference on Learning Representations*, 2024.

Nikola Jovanovic, Ismail Labiad, Tomas Soucek, Martin Vechev, and Pierre Fernandez. Watermarking autoregressive image generation. In *Advances in Neural Information Processing Systems*, 2025.

Denis Lukovnikov, Andreas Muller, Erwin Quiring, and Asja Fischer. ClusterMark: Towards robust watermarking for autoregressive image generators with visual token clustering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Yihan Wu, Xuehao Cui, Ruibo Chen, Georgios Milis, and Heng Huang. A watermark for auto-regressive image generation models. arXiv:2506.11371, 2025.

Yijian Lu, Aiwei Liu, Dianzhi Yu, Jingjing Li, and Irwin King. An entropy-based text watermarking detection method. In *Proceedings of ACL*, 2024.

Taehyun Lee, Seokhee Hong, Jaewoo Ahn, Ilgee Hong, Hwaran Lee, Sangdoo Yun, Jamin Shin, and Gunhee Kim. Who wrote this code? Watermarking for code generation. In *Proceedings of ACL*, 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. In *Advances in Neural Information Processing Systems*, 2024.

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. In *Advances in Neural Information Processing Systems*, 2024.

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. In *Advances in Neural Information Processing Systems*, 2024.

Yu Tong, Zihao Pan, Shuai Yang, and Kaiyang Zhou. Training-free watermarking for autoregressive image generation. arXiv:2505.14673, 2025.

Louis Kerner, Michel Meintz, Bihe Zhao, Franziska Boenisch, and Adam Dziedzic. BitMark: Watermarking bitwise autoregressive image generative models. In *Advances in Neural Information Processing Systems*, 2025.

Hai Ci, Pei Yang, Yiren Song, and Mike Zheng Shou. RingID: Rethinking Tree-Ring watermarking for enhanced multi-key identification. In *European Conference on Computer Vision*, 2024.

Liangqi Lei, Keke Gai, Jing Yu, and Liehuang Zhu. DiffuseTrace: A transparent and flexible watermarking scheme for latent diffusion model. arXiv:2405.02696, 2024.

Hai Ci, Yiren Song, Pei Yang, Jinheng Xie, and Mike Zheng Shou. WMAdapter: Adding watermark control to latent diffusion models. In *International Conference on Machine Learning*, 2025.

Lu Zhang and Liang Zeng. SAT-LDM: Provably generalizable image watermarking for latent diffusion models with self-augmented training. arXiv:2501.00463, 2025.

Tom Sander, Pierre Fernandez, Alain Durmus, Teddy Furon, and Matthijs Douze. Watermark Anything with localized messages. In *International Conference on Learning Representations*, 2025.

Vishal Asnani, John Collomosse, Tu Bui, Xiaoming Liu, and Shruti Agarwal. ProMark: Proactive diffusion watermarking for causal attribution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

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. In *International Conference on Learning Representations*, 2025.

Yunzhuo Chen, Jordan Vice, Naveed Akhtar, Nur Haldar, and Ajmal Mian. Dynamic watermarks in images generated by diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2025.

Yuzhuo Chen, Zehua Ma, Han Fang, Weiming Zhang, and Nenghai Yu. TAG-WM: Tamper-aware generative image watermarking via diffusion inversion sensitivity. In *IEEE/CVF International Conference on Computer Vision*, 2025.

Kasra Arabi, R. Teal Witter, Chinmay Hegde, and Niv Cohen. SEAL: Semantic aware image watermarking. In *IEEE/CVF International Conference on Computer Vision*, 2025.

Zilan Wang, Junfeng Guo, Jiacheng Zhu, Yiming Li, Heng Huang, Muhao Chen, and Zhengzhong Tu. SleeperMark: Towards robust watermark against fine-tuning text-to-image diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Yepeng Liu, Yiren Song, Hai Ci, Yu Zhang, Haofan Wang, Mike Zheng Shou, and Yuheng Bu. Image watermarks are removable using controllable regeneration from clean noise. In *International Conference on Learning Representations*, 2025.

Andreas Muller, Denis Lukovnikov, Jonas Thietke, Asja Fischer, and Erwin Quiring. Black-box forgery attacks on semantic watermarks for diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Qiran Lai and Adrian G. Bors. Vector Quantization-Based Watermarking for Autoregressive Generated Images. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2026.

Yepeng Liu and Yuheng Bu. Adaptive text watermark for large language models. In *International Conference on Machine Learning*, 2024.

Yuepeng Hu, Zhengyuan Jiang, Moyang Guo, and Neil Zhenqiang Gong. A Transfer Attack to Image Watermarks. In *International Conference on Learning Representations*, 2025.

Pierre Fernandez, Tomas Soucek, Nikola Jovanovic, Hady Elsahar, Sylvestre-Alvise Rebuffi, Valeriu Lacatusu, Tuan Tran, and Alexandre Mourachko. Geometric Image Synchronization with Deep Watermarking. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2026.

Andreas Muller, Denis Lukovnikov, Shingo Kodama, Minh Pham, Anubhav Jain, Jonathan Petit, Niv Cohen, and Asja Fischer. On the Robustness of Watermarking for Autoregressive Image Generation. arXiv:2604.11720, 2026.

Shreyas Dixit, Ashhar Aziz, Shashwat Bajpai, Vasu Sharma, Aman Chadha, Vinija Jain, and Amitava Das. PECCVAI: Overcoming the Brittleness of AI Image Watermarking Under Visual Paraphrasing Attacks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Zhenliang Gan, Chunya Liu, Yichao Tang, Binghao Wang, Shiwen Cui, Weiqiang Wang, and Xinpeng Zhang. GenPTW: Latent Image Watermarking for Provenance Tracing and Tamper Localization. In *AAAI Conference on Artificial Intelligence*, 2026.

A Detailed Protocols and Calibration

The main text reports compact protocol summaries; Table 5 collects the full split, calibration, and freezing rules in one place. The appendix then follows the main-paper logic: WMAR robustness details, ClusterMark supplementary diagnostics, and finally component/mechanism analyses, all under the same image-disjoint and frozen-selection protocol.

Algorithm 1 Offline training and online verification in RAV

Require: Frozen recovery module; training images with original verification symbols; independent negative calibration set.

Offline reliability training:

1: Recover each training image and construct the recovery-side feature field Φ.
2: Compute symbol-survival labels $y_t = \mathbf{1}[\hat{s}_t = s_t]$.
3: Train f_θ with the binary cross-entropy loss in Eq. (3).
4: Freeze the trained reliability predictor f_θ.

Online verification:

5: Recover $\hat{S}$ and Φ from the suspect image.
6: Compute $\mathbf{r} = \sigma(f_\theta(\Phi))$ without using the secret key.
7: Use key k to compute the keyed evidence $g_m(k)$.
8: Compose evidence weights $w_m = \Psi_m(r)$ according to the watermark structure.
9: Compute Z_{RAV} from Eq. (4) and compare it with the threshold τ frozen on the independent negative calibration set.

Table 5: Detailed protocols and freezing rules corresponding to the compact main-text summary. Transformed copies are grouped by original image ID.

Protocol	Data and freezing rule
RAR-XL WMAR grouped OOF	1,000 images and 9,000 attacked records; image-grouped five-fold prediction.
RAR-XL WMAR blind	1,000 newly generated images with a new key. The 200-image calibration split yields $200 \times 9 \times 64 = 115{,}200$ wrong-key null decisions for freezing the 1%/0.1% thresholds; the remaining 800 images form the one-time blind set.
RAR-XL true negatives / shifts	585,000 genuinely unwatermarked decisions; 28 frozen attack settings; six unseen CompressAI quality-3 models, all evaluated without retraining or threshold recalibration.
TAMING WMAR matched	1,000-image grouped OOF development; a separate 200-calibration/800-blind split, with 115,200 wrong-key null decisions used to freeze the 1%/0.1% thresholds; followed by 585,000 genuine-negative decisions, 28 frozen attack settings, and six unseen quality-3 neural codecs.
RAR-XL ClusterMark development	2,000 watermarked and 2,000 clean images with fixed 800/400/800 train/validation/test IDs; model selection uses validation data only.
RAR-XL ClusterMark blind	New prefix 7; 200 negative calibration images pooled over nine attacks yield 1,800 calibration scores. IDs 200–999 form the one-time blind set. Frozen-threshold reporting uses the 1% operating point; TPR at 0.1% FPR is reported as a held-out empirical ROC readout.
LlamaGen GPT-B matched	2,000 clean and 2,000 K64 images with fixed 800/400/800 IDs; eight pixel attacks and three regeneration attacks; no test-time model reselection.
GPT-B FID audit	50k clean and 50k K64 images compared with ImageNet val50k under frozen CFG$= 2.0$, top-$k = 0$ sampling.

A.1 Reliability-Probability Calibration Diagnostics

Beyond AUROC-based discrimination, we examine whether predicted reliabilities behave as meaningful probabilities. We report area under the precision–recall curve (AUPRC) to complement AUROC under class imbalance, and Brier score and expected calibration error (ECE) to assess probability calibration. Table 6 summarizes these diagnostics for RAR-XL ClusterMark. Logistic-regression and reliability-MLP results use the 800-image held-out test split (4,096,000 symbol positions), whereas the unary spatial CNN uses the separate 200-image validation-calibration split. The rows

therefore characterize probability quality within their respective protocols rather than serving as a direct cross-split ranking.

Table 6: Probability-level diagnostics for RAR-XL CLUSTERMARK cluster-survival prediction. Higher is better for AUROC and AUPRC; lower is better for Brier score and ECE.

Predictor	Evaluation split	AUROC ↑	AUPRC ↑	Brier ↓	ECE ↓
Logistic regression	Held-out test	.8404	.9010	.1567	.0076
Reliability MLP	Held-out test	.8424	.9023	.1556	.0033
Unary spatial CNN (main)	Validation-calibration	.8835	.9336	.1343	.0020

The low ECE values indicate that the learned reliability scores are not only discriminative but also closely aligned with empirical survival frequencies. This supports their use as probability-style evidence weights, while the independent threshold calibration used by the final detector remains unchanged.

A.2 IMPLEMENTATION DETAILS

RAV-WMAR. The reliability predictor operates on the 16×16 VQ-token grid. For each recovered token, the input features include the Top-8 codebook distances, relative quantization-gap information, soft-assignment statistics including entropy, a fixed projection of the pre-quantization latent, multi-view recovery-consistency cues, spatial coordinates, a token-ID representation, and the handcrafted quantization-margin prior. These key-independent features are concatenated into a spatial feature field and processed by a residual spatial CNN with 231,553 trainable parameters. The same reliability-network architecture is used for the RAR-XL and TAMING instantiations; only the generator/tokenizer-specific recovery front end is changed. The predictor is trained with the binary cross-entropy objective in Eq. (3), and its output is converted to the final WMAR evidence weight using $w_t^{\mathrm{W}} = \epsilon + (1 - \epsilon) r_t^{\mathrm{W}}$ with $\epsilon = 0.05$.

For RAR-XL, reliability development uses image-grouped five-fold out-of-fold evaluation over 1,000 images and nine canonical conditions, with all transformed versions of an image kept in the same fold. For TAMING, the same architecture is retained and the final reliability model is trained on the full 1,000-image development set for eight epochs, with the epoch count fixed from the preceding grouped cross-validation before the blind test. Neither the blind images nor their true-key scores are used for model selection.

RAV-ClusterMark. The Cluster Predictor provides a 64-dimensional posterior at each position. We use its log-probabilities together with five confidence/consistency scalars: maximum probability, top-two probability margin, inverse normalized entropy, hard-VQ cluster probability, and predictor/VQ agreement, giving a $69 \times 16 \times 16$ recovery-side feature field. The main reliability model is a unary spatial CNN with 39,585 trainable parameters. For architectural controls, we additionally use a pointwise MLP with 833 parameters and a pairwise spatial CNN with 41,665 parameters; the latter directly predicts transition survival rather than composing two endpoint reliabilities.

For the RAR-XL instance, the data are split by image ID into 800/400/800 train/validation/test images. The 400-image validation set is further divided into 200 images for model selection and 200 for calibration. The unary CNN is selected before test evaluation using the macro TPR@1% FPR over the nine validation attacks and is subsequently frozen. For the independent prefix-7 blind protocol, IDs 0–199 are used only for negative calibration and IDs 200–999 are read once for final evaluation.

For LlamaGen GPT-B, the same 64-class reliability interface is constructed from VQ pre-quantization features through the Token Predictor, K64 aggregation, and a frozen posterior adapter with temperature $T = 0.75$ and hard-VQ anchor $\beta = 2$. Reliability-model selection again uses only the training and validation partitions; the selected unary CNN reaches its best validation checkpoint at epoch 11, while the test split is never used for checkpoint selection. All reliability models are frozen before the corresponding watermark-verification evaluation.

B Additional WMAR Results

B.1 RAR-XL Reproduction

We first verify the baseline-aligned RAR-XL pipeline so that all later detector comparisons rest on a reproduced reference implementation. The feature/architecture ablation used to motivate the RAV-WMAR design is deferred to Appendix D together with the other mechanism analyses.

Table 7: Our official-pipeline reproduction on 1,000 RAR-XL images. Values are produced by our run rather than copied from the baseline paper.

Attack group	TPR
None	0.9990
Value transformations	0.8845
Geometric transformations	0.8310
DiffPure	0.7930

B.2 RAR-XL Robustness Details

We next expand the main blind evaluation along attack severity, selected difficult settings, and unseen neural compression. The severity plot shows how the detector changes as attacks become stronger; the selected sweep table exposes representative hard cases; and the neural-compression table tests a qualitatively different, unseen distortion family without retraining.

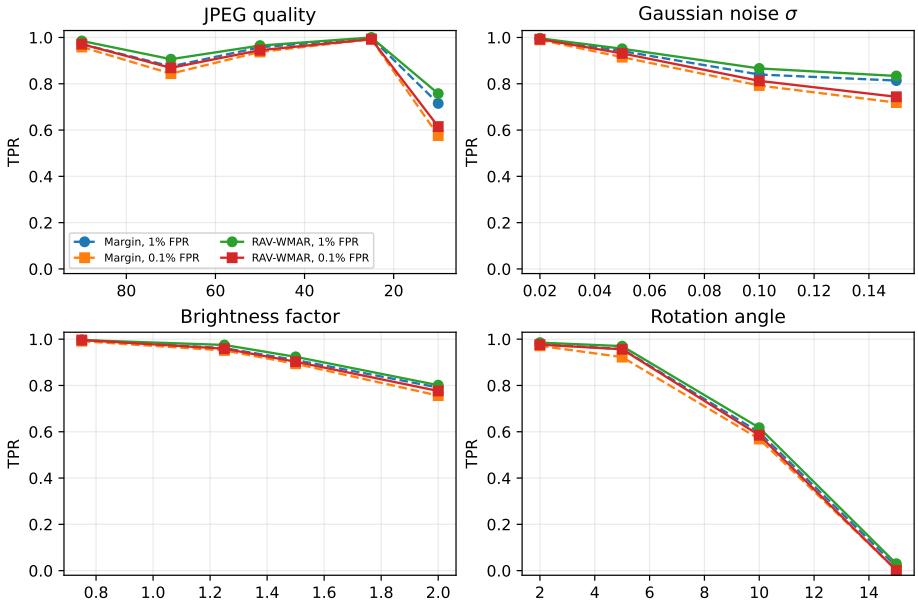

Figure 6: RAR-XL WMAR attack-severity lines for the handcrafted margin and RAV-WMAR. Reliability weighting generally preserves or improves TPR as attacks become stronger, but cannot restore evidence destroyed by severe geometric desynchronization.

Table 8: Frozen 800-image attack sweep: selected difficult conditions. Positive deltas favor RAV-WMAR.

	TPR@1% FPR			TPR@0.1% FPR		
Setting	Margin	RAV	Δ	Margin	RAV	Δ
JPEG Q=10	.7150	.7575	+.0425	.5763	.6150	+.0387
JPEG Q=70	.8750	.9063	+.0313	.8438	.8688	+.0250
Noise $\sigma = .10$	.8400	.8663	+.0263	.7925	.8125	+.0200
Noise $\sigma = .15$	.8138	.8338	+.0200	.7188	.7438	+.0250
Rotation 5°	.9575	.9700	+.0125	.9225	.9563	+.0338
Rotation 10°	.5975	.6175	+.0200	.5688	.5850	+.0163
Rotation 15°	.0175	.0300	+.0125	.0025	.0013	-.0013
Crop .80 + JPEG 50	.0700	.0913	+.0213	.0200	.0200	0

Table 9: Unseen neural compression on the frozen 800-image RAR-XL blind set.

Compressor	TPR@1% FPR				TPR@0.1% FPR			
	Equal	Margin	RAV	Perm.	Equal	Margin	RAV	Perm.
BMSHJ18 Factorized	.9413	.9650	.9638	.8938	.8938	.9213	.9400	.8138
BMSHJ18 Hyperprior	.9438	.9600	.9650	.9200	.9063	.9350	.9375	.8600
Cheng20 Anchor	.9413	.9588	.9650	.9125	.9125	.9275	.9363	.8488
Cheng20 Attention	.9563	.9625	.9725	.9188	.9188	.9413	.9513	.8625
MBT18 Scale	.9463	.9713	.9775	.9063	.9013	.9375	.9513	.8450
MBT18	.9638	.9713	.9800	.9288	.9238	.9475	.9525	.8775
Mean	.9488	.9648	**.9706**	.9133	.9094	.9350	**.9448**	.8513

B.3 TAMING RAV-WMAR Cross-Generator Extension

The TAMING study independently re-instantiates the same RAV-WMAR principle on a different generator/tokenizer pipeline. We report the matched OOF/blind comparison first, followed by unseen neural compression and the frozen-protocol follow-up summary.

Table 10: Matched TAMING RAV-WMAR results. OOF is image-grouped five-fold evaluation on 1,000 development images; Blind is the independently generated 800-image one-time test after 200-image wrong-key calibration.

Method	OOF TPR@1% FPR	OOF TPR@0.1% FPR	Blind Overall@1%	Blind Overall@0.1%
Equal	.8310	.7930	.8282	.7849
Handcrafted margin	.8440	.8106	.8451	.8072
RAV-WMAR	**.8564**	**.8297**	**.8604**	**.8304**
Permuted RAV	.7961	.7354	.7908	.7294

On grouped OOF, RAV-WMAR improves the handcrafted margin by 0.0124/0.0191 at the two FPR operating points. On the one-time blind split, the gains are 0.0153 (95% CI [0.0119, 0.0186]) and 0.0232 ([0.0193, 0.0271]). The blind wrong-key FPR for RAV-WMAR is 0.01039/0.00089. With the same frozen model and thresholds, genuinely unwatermarked TAMING data yield FPR 0.01075/0.00099 over 585,000 decisions; the 28-setting sweep gives 0.8187/0.7883 macro TPR versus 0.8028/0.7685 for the margin; and six unseen neural codecs give 0.8673/0.7908 versus 0.8250/0.7465 for the margin.

Table 11: Matched TAMING unseen neural compression on the frozen 800-image blind split. All six CompressAI models use quality 3; positive deltas in the text are measured against the handcrafted margin.

Compressor	TPR@1% FPR				TPR@0.1% FPR			
	Equal	Margin	RAV	Perm.	Equal	Margin	RAV	Perm.
BMSHJ18 Factorized	.6938	.7263	**.7863**	.6025	.5625	.6075	**.6650**	.4563
BMSHJ18 Hyperprior	.7963	.8413	**.8750**	.7563	.7225	.7513	**.7950**	.6488
Cheng20 Anchor	.7825	.8225	**.8650**	.7413	.7000	.7438	**.7963**	.6313
Cheng20 Attention	.8413	.8663	**.9063**	.7863	.7638	.8000	**.8413**	.7038
MBT18 Scale	.8100	.8375	**.8763**	.7688	.7425	.7888	**.8175**	.6650
MBT18	.8325	.8563	**.8950**	.7950	.7588	.7875	**.8300**	.6888
Mean	.7927	.8250	**.8673**	.7417	.7083	.7465	**.7908**	.6323

Across these six codecs, RAV exceeds the handcrafted margin by 0.0423 at 1% FPR (95% CI [0.0344, 0.0504]) and 0.0444 at 0.1% FPR ([0.0367, 0.0525]); the gain is positive for every codec at both operating points. The frozen-threshold NC diagnostic FPR for RAV is 0.01040/0.00095.

Table 12: TAMING frozen-protocol follow-up summary. The 28-setting row is a macro average over the same attack strengths/combinations used for RAR-XL; true-negative FPR uses 585,000 decisions.

Follow-up metric	Equal	Margin	RAV
28-setting macro TPR@1% FPR	.7911	.8028	**.8187**
28-setting macro TPR@0.1% FPR	.7515	.7685	**.7883**
Genuine-unwatermarked FPR@1%	.00949	.01040	.01075
Genuine-unwatermarked FPR@0.1%	.00097	.00111	.00099

The 28-setting RAV-minus-margin sign counts are 26/1/1 (positive/tie/negative) at 1% FPR and 26/0/2 at 0.1% FPR. These follow-ups reuse the one-time blind images only as a frozen robustness substrate and are not presented as additional independent blind tests.

C Additional ClusterMark Results

C.1 Uncertainty on the Prefix-7 Blind Evaluation

To quantify uncertainty in the independent RAR-XL prefix-7 blind evaluation of Table 4, we perform image-paired bootstrap comparisons between the validation-selected unary RAV model and representative baselines. Table 13 reports the mean paired differences and their 95% confidence intervals. Positive values indicate an improvement of the unary RAV model.

Table 13: Image-paired bootstrap uncertainty for the RAR-XL prefix-7 blind evaluation. Entries are mean differences (unary RAV minus comparator) with 95% confidence intervals.

Comparator	Δ AUROC	Δ Emp. TPR@1% FPR	Δ Emp. TPR@0.1% FPR
Hard Cluster Predictor	$+.0113$ $[.0095, .0131]$	$+.0468$ $[.0376, .0571]$	$+.0827$ $[.0581, .1008]$
Pointwise MLP	$+.0018$ $[.0013, .0023]$	$+.0053$ $[.0000, .0099]$	$+.0098$ $[.0018, .0181]$
Permuted unary	$+.0184$ $[.0160, .0209]$	$+.0916$ $[.0775, .1058]$	$+.1302$ $[.1088, .1467]$

The confidence intervals confirm that the main prefix-7 gains over the hard Cluster Predictor and the spatial-permutation control are robust to image-level resampling. The improvement over the pointwise MLP is substantially smaller, consistent with the interpretation that spatial context provides an additional but comparatively modest benefit beyond occurrence-specific reliability prediction.

C.2 LLAMAGEN GPT-B CROSS-GENERATOR DIAGNOSTICS

Following the WMAR robustness details, we next provide the supplementary ClusterMark evidence. Because the GPT-B study requires a locally reconstructed K64 generation pipeline, we additionally verify its generation quality against the original ClusterMark reference. This is a reproduction sanity check rather than a detector metric; RAV changes verification only, so every detector variant shares the same K64 generation path and quality cost.

Table 14: LlamaGen GPT-B six-attack macro diagnostics. The macro averages JPEG, Blur, Gaussian noise, Salt-and-pepper, Color Jitter, and Regeneration and excludes Clean.

Detector	AUROC	TPR@1% FPR
K64 hard VQ	.9122	.5560
K64 + Token Predictor	.9050	.5421
K64 + Cluster Predictor, hard	.9038	.5021
Posterior soft	**.9202**	**.5787**
Max probability	.9169	.5717
Pointwise MLP	.9185	.5711
Pairwise spatial CNN	.9171	.5717
RAV-ClusterMark, unary CNN	.9181	.5777
Permuted unary control	.9181	.5761

Table 15: Separate protocol-matched 50k-image GPT-B generation-path sanity check for the locally reconstructed K64 pipeline. This reproduction audit is not a detector metric: RAV changes verification only, so K64 generation quality is shared by all detector variants.

Generation condition	Local CleanFID	ClusterMark reference	Local − reference
GPT-B clean	6.0202	6.01	+0.0102
GPT-B + K64	6.1025	6.12	-0.0175
K64 − clean	+0.0823	+0.11	–

The official clean and custom K64 generation paths were additionally checked with a $\delta = 0$ exact-equivalence gate: under the frozen CFG=2.0/top-$k = 0$ sampling schedule, class mismatch, pixel mismatch, and maximum absolute pixel difference are all zero when watermark bias is disabled.

D ADDITIONAL ABLATION AND MECHANISM DIAGNOSTICS

D.1 RAV-WMAR COMPONENT AND ARCHITECTURE ABLATION

After the detailed robustness results for both watermark families, we collect the remaining mechanism-focused evidence. Table 16 reports the image-grouped RAV-WMAR feature/architecture ablation that motivates the frozen reliability design used in the main experiments.

Table 16: WMAR image-grouped five-fold OOF feature and architecture ablation. Deltas are relative to the handcrafted-margin baseline.

Variant	TPR@1% FPR	TPR@0.1% FPR	Δ@1%	Δ@0.1%
Equal	.8714	.8390	-.0137	-.0181
Handcrafted margin	.8851	.8571	0	0
Margin only	.8851	.8571	0	0
Top-K only	.8959	.8692	+.0108	+.0121
Margin + Top-K	.8963	.8691	+.0112	+.0120
Margin + Top-K + ID	.8942	.8684	+.0091	+.0113
Margin + Top-K + ID + latent	.8949	.8699	+.0098	+.0128
Full pointwise	.8887	.8620	+.0036	+.0049
Full spatial	.8948	.8683	+.0097	+.0112

D.2 Cross-Attack Reliability Diagnostics

Figure 7 then provides two complementary checks on the reliability mechanism: the left panel summarizes RAV-WMAR gains across the 28 frozen attack settings, while the right panel tests ClusterMark survival prediction under leave-one-attack-out training.

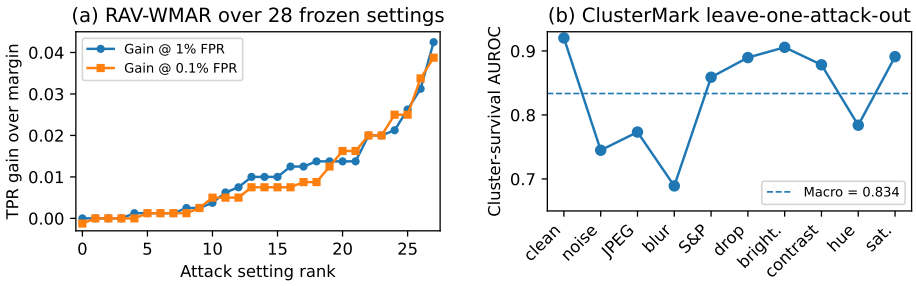

Figure 7: Diagnostics for reliability validity. Left: independently sorted RAV-WMAR gains over the handcrafted margin across the 28 frozen RAR-XL attack settings at the two operating points. Right: RAR-XL ClusterMark cluster-survival AUROC under leave-one-attack-out training.