COSIGNATURE: CROSS-MODALITY WATERMARKING WITH VERIFIABLE BINDING

Anonymous authors
Paper under double-blind review

ABSTRACT

The rapid deployment of multimodal models has enabled the seamless generation of interleaved text and images. However, existing watermarking schemes treat these modalities in isolation, leaving systems vulnerable to misinformation campaigns where malicious actors manipulate context or swap single-modality content. To address this, we propose CoSignature, a cross-modality watermarking scheme that establishes a verifiable binding between visual and textual outputs via a shared content ID, which we hope to embed into both modalities. To overcome the limited capacity of text watermarking, we employ a novel "zero-bit" binding mechanism: rather than embedding the content ID as a payload, we use it to seed the text generation algorithm itself, converting binding verification into a statistical detection task that disambiguates the true content ID from noisy candidates retrieved by the image decoder. We provide theoretical guarantee that our binding error probability decays exponentially with text length. Empirical evaluation on four distinct architectures—native unified models (Janus-Pro-7B), composite pipelines (Qwen3-8B + Qwen-Image), Qwen3-VL-8B, and BAGEL-7B-MoT—demonstrates the efficacy of our approach, achieving $\approx 98\%$ binding accuracy on clean data, $> 95\%$ accuracy in detecting swap attacks.

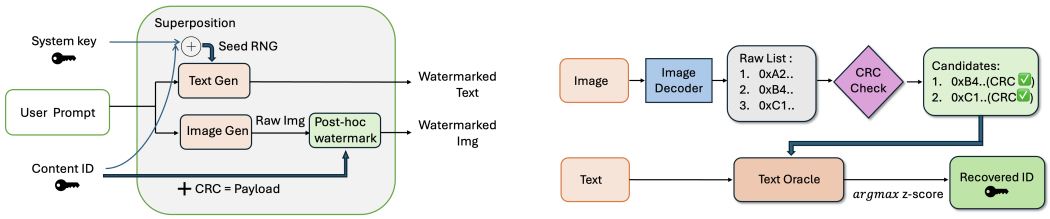
1 INTRODUCTION

The rapid evolution of Multimodal Large Language Models (MLLMs) has democratized the creation of mixed-media content, enabling the seamless generation of interleaved text and imagery (Zhang et al., 2025; Chen et al., 2025; Wang et al., 2025; Liu et al., 2025b; Deng et al., 2025; Hurst et al., 2024; Ge et al., 2024; Xie et al., 2024). Figure 3 in Appendix displays such a generation. While this capability fosters creativity, it exacerbates the risks of *compositional misinformation*: bad actors can separately generate semantically consistent images and text, then combine them into misleading articles. For instance, during a crisis such as the 2025 LA wildfires, AI-generated images of exaggerated destruction could be paired with separately AI-generated text to fabricate a false news report. Both components are individually plausible and would pass independent watermark verification, yet the combination is forged. Establishing the provenance of such content requires not only verifying the source of each modality but ensuring that the image and text were generated *together* as a coherent semantic unit.

Crucially, cross-modal binding protects the *platform* as much as the public. Without binding, independent watermarks confirm that both components originated from the platform, but the platform has no way to prove it did not generate them together, which falsely implicates it as the source of the entire narrative. With binding, the platform can prove "we generated these components, but not together". This can distinguish independent generation from joint authorship. Our intended application is joint provenance of multimodal *articles* (e.g., news articles, blog posts, reports with accompanying images).

Current watermarking safeguards, however, remain largely siloed. State-of-the-art image watermarking (Fernandez et al., 2024; Sander et al., 2024; Petrov et al., 2025; Souček et al., 2025) and text watermarking (Kirchenbauer et al., 2023a; Dathathri et al., 2024; Zhao et al., 2023; Aaronson, 2022) operate as disjointed layers. A standard "pipeline" defense might independently verify that an image is AI-generated and the associated text is also AI-generated, but it cannot mathematically prove that *this* specific text is associated with *this* specific image. This "binding gap" renders current systems vulnerable to swap attacks, where attackers strip or replace single modality content to break

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.



(a) **Generation.** (b) **Verification.**

Figure 1: **Overview of CoSignature. (a) Generation:** A dual-key strategy: the Content ID and the system key seeds the text generation; and the Content ID is injected as the image payload. **(b) Verification:** The image decoder outputs noisy candidates. The Text Oracle acts as a selector, using the text's watermark statistics to recover the correct Content ID from the noise.

the semantic integrity of the content while bypassing detection. While preliminary works such as Jovanović et al. (2025) have attempted to bridge these modalities via joint watermarking of interleaved content, they are restricted to zero-bit detection on native autoregressive models, leaving the challenge of robust, multi-bit binding unresolved.

In this paper, we propose CoSignature, a unified multi-bit watermarking scheme designed to establish a verifiable binding between visual and textual outputs. We employ a novel superposition watermarking scheme, where the text stream carries a dual signal: a global "System Key" for independent provenance and a "Content Key" mathematically bound to the unique Content ID. This design secures the semantic integrity of the pair, enforcing a cryptographic dependency that prevents adversarial content swapping, as shown by the binding verification in Figure 3.

By explicitly formalizing this cross-modal interaction, we answer a fundamental question: *Can information from one modality substitute for lost redundancy in another?* We demonstrate that the answer is yes, and we measure the exchange rate. CoSignature couples a Soft-Ordered Statistics Decoder (Soft-OSD) over the image payload with a Text Oracle that ranks the surviving candidates, and the two components contribute differently. The soft-decision search is what lifts payload recovery under distortion relative to a hard-decision decoder; at a matched false-accept rate it is never worse than an idealized channel code given the same redundancy, and gains up to 14.3 points under severe cropping (Sec. C.8). The Text Oracle contributes something no channel code can: it scores candidates using a statistic carried by the *other* modality, and therefore substitutes for checksum redundancy directly. Shrinking the checksum from 32 bits to 8 leaves a conventional decoder unable to identify a payload at all, while the oracle still attains $\approx 98\%$ of the achievable recovery rate—returning 24 bits to Content ID capacity (Sec. C.9). It is also the only mechanism here that can verify *binding*: whether a given text and image were generated together, a question no amount of image-payload redundancy can answer.

To summarize, our contributions are as follows:

1. We establish the first formal definition for cross-modality watermarking, that supports three detection scenarios: (1) *Independent Detection*, proving provenance from a single modality in isolation; (2) *Binding Detection*, verifying that the full image-text pair was generated as a coherent unit; and (3) *Swap Detection*, identifying inconsistencies where valid watermarked content is spliced with mismatched or external data.

2. We propose CoSignature, a unified protocol that mathematically couples visual and textual outputs. Our approach overcomes the bandwidth bottleneck of short-text generation by converting the *multi-bit embedding problem* into a scalable *zero-bit verification problem* via a dual-key superposition scheme. Crucially, this framework is *highly modular*: it is designed to wrap around any standard image watermarking scheme and any sampling-based LLM watermarking scheme, enabling deployment on top of existing proprietary safeguards without architectural changes.

3. We introduce a joint source–channel decoding paradigm in which the text modality acts as a side-channel oracle over the candidate set produced by the image decoder, and we quantify what each component contributes. Against an image-only baseline whose payload is protected by an *idealized* channel code at matched redundancy *and* matched false-accept rate, the joint decoder is never worse and gains up to 14.3 points under severe distortion. Separately, because the oracle ranks candidates

on a statistic no checksum can access, it substitutes for checksum redundancy: an 8-bit check retains $\approx 98\%$ of the recovery rate achievable with a 32-bit CRC, returning 24 bits to Content ID capacity. We theoretically prove that the binding error probability decays exponentially with text length, which guarantees the text oracle accuracy.

4. We evaluate CoSignature across four distinct architectures: the native autoregressive model (Janus-Pro-7B), the composite pipeline (Qwen-Pipeline), Qwen3-VL-8B, and BAGEL-7B-MoT. The latter additionally demonstrates modularity by instantiating CoSignature with PixelSeal as a drop-in replacement for VideoSeal. Our extensive evaluation demonstrates that CoSignature achieves a Binding True Positive Rate (TPR) of 98.7% on Janus-Pro and 97.4% on Qwen, with a fixed 1.0% False Positive Rate. Furthermore, it accurately detects swap attacks with $> 95\%$ accuracy, effectively preventing context-shifting manipulation.

2 Related Work

LLM Text Watermarking typically employs logit-bias techniques during autoregressive generation to embed signals in the token distribution (Kirchenbauer et al., 2023a;b; Dathathri et al., 2024; Zhao et al., 2023; Aaronson, 2022; Shi et al., 2024; Zhao et al., 2024a; Christ et al., 2024; Zhao et al., 2024b; Wu et al., 2024; Qu et al., 2024). Kirchenbauer et al. (2023a) proposed to split the entire token space into green and red lists through a pseudo-random hashing function controlled by previous tokens. The injection boosts the generation of green tokens and the detection is achieved via statistical hypothesis testing. Zhao et al. (2023) later proposed to simplify the aforementioned approach by using a constant red-green split can make the watermark twice robust to edits, provably. Kuditipudi et al. (2023) and Dathathri et al. (2024) also proposed variants that are distortion free and scalable to use in production systems. As opposed to biasing the token level distribution, another line of work embeds watermark signal into the semantic space to make watermark robust to paraphrasing (Hou et al., 2024a;b; Liu et al., 2023; Ren et al., 2024). As for post-hoc text watermark, Chang et al. (2024) proposed PostMark that injects a set of "input dependent words" (calculated from their semantic embeddings) into the generated text. This approach enables text watermark without accessing LLM's logits and allows the watermark to be more robust to paraphrasing.

Image Watermarking has gravitated toward "post-hoc" pixel-space injections (Fernandez et al., 2024; Sander et al., 2024; Petrov et al., 2025; Souček et al., 2025) as they decouple image generation from watermark addition while preserving visual fidelity and robustness against geometric distortions. In parallel, generation-time watermarking methods (Fernandez et al., 2023; Yu et al., 2020) where watermark information is embedded by modifying the image generation process itself as well as diffusion-based latent-space injection approaches (Wen et al., 2023; Zhang et al., 2024a) have emerged as promising alternatives due to their increased resistance to watermark removal attacks. However, these watermarking schemes often incur substantial computational overhead or introduce noticeable visual artifacts, limiting their practical applicability and leading post-hoc pixel-space approaches to dominate industrial use cases (Fernandez et al., 2024; Gowal et al., 2025).

Cross-modality Watermarking. VLA-Mark (Liu et al., 2025a) preserves multimodal coherence but applies text-only watermarking with no image signal or binding mechanism. Jovanović et al. (2025) initiated zero-bit cross-modality watermarking on native autoregressive models, but performs only source attribution (not multi-bit recovery), lacks swap detection, and is restricted to unified AR architectures. Prior audio-video cross-modal work Zhang et al. (2024b) does not extend to text-image settings. In contrast, CoSignature embeds a multi-bit content ID, establishes cryptographic binding, and ensures independent verifiability. Table 1 summarizes the key differences. Standards such as C2PA (C2PA) bind cryptographic signatures to file metadata but do not survive format conversion and provide no text provenance.

3 Methodology

A robust cross-modal watermark must satisfy two critical conditions: (1) *Binding*, where the presence of both modalities cryptographically validates their pairing; and (2) *Independence*, where the removal of one modality does not compromise the provenance of the remaining content.

Challenges of cross-modal binding. Establishing this link is non-trivial due to the asymmetric information capacity of the two modalities. Cryptographic binding requires a high-entropy content ID (e.g., 224 bits) to prevent collisions, but current multi-bit text watermarking requires $\sim$10 tokens

Table 1: Qualitative Comparison of Provenance Methods. CoSignature is the first method to provide multi-bit cross-modal binding with swap detection, independent of model architecture.

Method	Multi-bit Payload	Text Prov.	Image Prov.	Binding / Swap Det.	Architecture Requirement
C2PA	✓†	✗	✓†	✗	Any (metadata)
VLA-Mark	✗	✓*	✗	✗	VLMs
Jovanovic et al.	✗	✓*	✓*	✗	Unified AR only
Indep. WM	✓	✓	✓	✗	Any
CoSignature	✓	✓	✓	✓	Any

†Metadata-based; not embedded in content, easily stripped. *Zero-bit detection only.

per bit (Wang et al., 2024), demanding >2000 tokens—far exceeding typical outputs. Post-hoc text steganography (synonym substitution, Unicode manipulation) is similarly fragile to re-encoding and lacks statistical guarantees. This capacity gap is why no straightforward "embed the same ID in both modalities" baseline exists. An alternative of embedding a hash of the text into the image breaks upon any character edit and prohibits independent detection when one modality is lost. Any viable solution must simultaneously support binding verification and independent single-modality verification.

To overcome these limitations, we propose a novel binding mechanism that shifts the paradigm from *payload extraction* to *signal detection*. Instead of embedding the Content ID *into* the text as a message, we use the Content ID to *seed* the text watermarking algorithm itself. This transforms the prohibitive multi-bit payload requirement into a scalable "Zero-Bit" detection task: the Text Oracle need only verify the *presence* of the statistical pattern generated by the Content ID, rather than decoding a complex sequence. This allows CoSignature to function robustly even on short text sequences where traditional multi-bit embedding is mathematically impossible.

CoSignature is built upon established single-modality watermarking techniques, focusing on the novel binding protocol that couples these separate channels via a shared Content ID and a joint source-channel decoder. The framework is highly modular: it can instantiate any standard zero-bit LLM watermarking technique (e.g., SynthID (Dathathri et al., 2024), KGW (Kirchenbauer et al., 2023a)) by simply modifying the initialization of the pseudo-random number generator (PRNG) seeds. Similarly, it supports any high-capacity image watermarking scheme capable of embedding the required payload. For simplicity, we employ the KGW watermarking (Kirchenbauer et al., 2023a) for text, and a post-hoc watermarking (e.g., VideoSeal (Fernandez et al., 2024)) for the visual output. Figure 1 illustrates the complete pipelines. We discuss the extension to Gumbel-based watermarking Aaronson (2022) in Appendix C.6.

3.1 Coupling the Channels: Watermark Embedding

We define our payload as a tuple $\mathcal{P} = (k_{cont}, \mathcal{C})$, where $k_{cont} \in \{0,1\}^B$ is a randomly generated B-bit content ID (e.g., 224 bits) and $\mathcal{C} = \text{CRC}(k_{cont})$ is a C-bit Cyclic Redundancy Check (e.g., 32 bits) used for fast candidate filtering. The total payload size is $L = B + C$ bits.

We embed the full payload $\mathcal{P}$ into the image I using a high-capacity post-hoc watermarking network: $I_w = \text{Embed}_{img}(I, \mathcal{P})$. The image carrier provides the primary "noisy" storage for the content information. For the text modality, we do not embed the payload as a sequence of bits. Instead, we use the content ID k_{cont} to seed the generation process itself. We introduce a **dual-key superposition scheme** that applies logit biases based on the interaction between a static system key (k_{sys}) and the dynamic content key (k_{cont}). At each generation step t, with previous context $x_{<t}$, we derive two independent pseudo-random seeds: $s_{sys}^{(t)} = \text{Hash}(k_{sys}, x_{<t}), \quad s_{cont}^{(t)} = \text{Hash}(k_{cont}, x_{<t})$. Using these seeds, we permute the vocabulary $\mathcal{V}$ to define two independent "Green Lists" (subsets of the vocabulary), each containing a fraction γ of the total tokens:

$$\mathcal{G}_{sys}^{(t)} = \text{PRG}(s_{sys}^{(t)})[:\gamma|\mathcal{V}|], \quad \mathcal{G}_{cont}^{(t)} = \text{PRG}(s_{cont}^{(t)})[:\gamma|\mathcal{V}|].$$

We then apply a logit bias δ to any token appearing in these lists. The final logit l_v for token v is modified as:

$$l'_v = l_v + \delta \cdot (\mathbb{I}[v \in \mathcal{G}_{sys}^{(t)}] + \mathbb{I}[v \in \mathcal{G}_{cont}^{(t)}]), \tag{1}$$

where $\mathbb{I}[\cdot]$ is the indicator function. This superposition creates three distinct token partitions: (1) Red tokens (bias 0): $v \notin \mathcal{G}_{sys} \cup \mathcal{G}_{cont}$; (2) Single-green tokens (bias δ): v is in exactly one list; (3) Double-green tokens (bias 2δ): $v \in \mathcal{G}_{sys} \cap \mathcal{G}_{cont}$.

This structure is crucial for our threat model: k_{sys} allows for text-only verification of model ownership, while k_{cont} serves as the secret binding link that can only be verified via the joint decoder.

3.2 JOINT SOURCE-CHANNEL DECODER

Recovering the content ID from a distorted image is challenging due to bit errors. We propose a Soft-Ordered Statistics Decoder (Soft-OSD) enhanced by a Text Oracle. This pipeline treats the text watermark as a side-channel to error-correct the image watermark. The decoding process follows three stages:

Step 1: Soft Demodulation. Given a query image $\hat{I}$, we extract the soft log-probabilities (logits) for the L-bit payload using the image watermark detector: $\mathbf{y} = \text{Detect}_{img}(\hat{I}) \in \mathbb{R}^L$. A hard decision vector $\hat{\mathbf{b}} = \text{sgn}(\mathbf{y})$ gives the baseline estimate. Due to image distortions (e.g., JPEG, cropping), $\hat{\mathbf{b}}$ typically contains bit errors that corrupt the payload.

Step 2: OSD Candidate Generation. To correct these errors, we identify the "weakest" bits in the payload. Let $\mathcal{W}$ be the set of indices corresponding to the D bits with the lowest confidence magnitudes $|y_i|$. Instead of an intractable brute-force search over the full space 2^L, we restrict our search to the subspace defined by $\mathcal{W}$. We generate a set of candidate vectors $\mathbf{C}$ by enumerating all 2^D binary patterns over the indices in $\mathcal{W}$, while fixing the $L - D$ high-confidence bits.

For every candidate vector $c \in \mathbf{C}$, we first perform a CRC validity check. If valid, we parse the vector to retrieve the candidate Content ID:

$$\mathcal{K}_{cand} = \{c[1:B] \mid c \in \mathbf{C}, \text{CRC}(c) = \text{Valid}\} \tag{2}$$

This step filters the 2^D permutations down to a small set of valid IDs $\mathcal{K}_{cand}$ (typically $|\mathcal{K}_{cand}| \leq 5$). The CRC bits are discarded, and the surviving IDs are passed to the Text Oracle for final verification.

Step 3: Text Oracle Verification. To maximize detection power, we construct a *weighted scoring function* $E(k)$ that ranks the candidate keys $\mathcal{K}_{cand}$ generated by the OSD step. Given a text sequence $T = (x_1, \ldots, x_N)$, we define the score for a specific candidate k as the sum of weighted token hits, where the weights correspond to the magnitude of the logit bias (δ vs. 2δ) predicted by that key:

$$E(k) = \sum_{t=1}^{N} \Big(2 \cdot \mathbb{I}[x_t \in \mathcal{G}_{sys}^{(t)} \cap \mathcal{G}_{cont}^{(t)}(k)] + 1 \cdot \mathbb{I}[x_t \in \mathcal{G}_{cont}^{(t)}(k) \setminus \mathcal{G}_{sys}^{(t)}] \Big). \tag{3}$$

This formulation assigns double weight to "double-green" tokens (which carry the strongest evidence of the specific binding) while still accumulating evidence from tokens that only hit the single green partition.

Finally, we identify the correct Content ID $\hat{k}_{cont}$ by maximizing the standardized z-score, which normalizes for sequence length and expected random overlap:

$$\hat{k}_{cont} = \underset{k \in \mathcal{K}_{cand}}{\text{argmax}} \frac{E(k) - \mu_w}{\sigma_w}, \tag{4}$$

where μ_w and σ_w represent the theoretical mean and standard deviation of the weighted sum under the null hypothesis, conditioned on the observed system green token fraction $\hat{p}_{sys} = N_{sys}/N$ (where N is the text length, equivalently denoted n in the scoring function):

$$\mu_w = N\gamma(1 + \hat{p}_{sys}), \quad \sigma_w = \sqrt{N\gamma(1-\gamma)(1 + 3\hat{p}_{sys})}. \tag{5}$$

This conditioning accounts for the fact that the system watermark biases token selection toward $\mathcal{G}_{sys}$, which affects the expected overlap with any candidate content key's green list (derivations provided in Appendix A).

Remark 3.1. The null hypothesis for binding verification assumes that the candidate content key k is uncorrelated with the generated text. This holds even though the text carries the system watermark, because the content key's green list $\mathcal{G}_{cont}(k)$ is seeded independently from the system key k_{sys}. For any false candidate $k' \neq k^*$, the pseudo-random partition $\mathcal{G}_{cont}(k')$ is statistically independent of both the system green list and the token choices, ensuring the null distribution remains valid.

3.3 Detection Regimes and Decision Criteria

To rigorously evaluate the robustness of CoSignature, we define criteria for three distinct detection regimes:

Independent Verification. It requires that a single modality can be independently verified that it is generated by our model assuming the other is lost. For image, this will solely depends on the applied image watermarking extractor. For text, we compute the standard z-score on the system-seeded green list partition $z_{sys} = \frac{N_{sys} - n\gamma}{\sqrt{n\gamma(1-\gamma)}}$, where $N_{sys} = \sum_{t=1}^{n} \mathbb{I}[x_t \in \mathcal{G}_{sys}^{(t)}]$ is the observed green token count. We classify the text as watermarked if $z_{sys} > \tau_{text}$, where the threshold τ_{text} is determined by the target false positive rate.

Binding Detection. For an input pair (I, T), let $\mathcal{K}_{cand}(I)$ be the set of M payload candidates recovered from the image. The binding score is $S_{binding}(I, T) = \max_{k \in \mathcal{K}_{cand}(I)} z(T, k)$, where $z(T, k)$ is the z-score derived from $E(k)$ (Eq. 3). To maintain a constant false positive rate α over M candidates, we apply a Bonferroni-corrected threshold $\tau_{bind}(M) = \Phi^{-1}(1 - \alpha/M)$. We accept the pair as *bound* if $S_{bind} > \tau_{bind}(M)$.

Swap Detection. Given an image-text pair (I, T), swap detection identifies when the image and text were not generated together, which means at least one modality can pass the independence detection, but the binding verification fails. We claim swap detected if any of the following conditions hold: (i) $(\mathcal{K}_{cand} \neq \emptyset) \wedge (S_{binding}(I, T) \leq \tau_{bind})$, (ii) $(z_{sys}(T) > \tau_{text}) \wedge (\mathcal{K}_{cand} = \emptyset)$, (iii)$(\mathcal{K}_{cand} \neq \emptyset) \wedge (z_{sys}(T) \leq \tau_{text})$ Condition (i) covers the WM-WM swap scenario where both modalities are watermarked but from different sessions. Condition (ii) detects when valid watermarked text is paired with an unwatermarked image. Condition (iii) detects when a watermarked image is paired with unwatermarked text.

3.4 Theoretical Analysis

We formally analyze the reliability of the cross-modal binding mechanism. Our core result proves that the Text Oracle acts as an asymptotic filter, where the probability of selecting a false candidate decays exponentially with text length.

Theorem 3.2 (Text Oracle Accuracy)**.** *Let N be the text length, M be the size of the candidate set $\mathcal{K}_{cand}$ containing the true key k^*. Assume there exists a deterministic constant $\Delta_{min} > 0$ such that the per-step expected signal margin satisfies $\mathbb{E}[Y_t \mid H_{t-1}] \geq \Delta_{min}$ almost surely for all t, where $Y_t = X_t^{(k^*)} - X_t^{(k')}$ is the score difference between the true key and any false key. The probability of detection error P_{error} is bounded by: $P_{error} \leq (M-1) \cdot \exp\left(-\frac{N \cdot \Delta_{min}^2}{8}\right)$.*

Proof. We model the cumulative score difference as a Doob Martingale and apply Azuma's Inequality with the deterministic threshold $\epsilon = N\Delta_{min}$. The existence of $\Delta_{min} > 0$ is guaranteed as long as the watermark injects a positive logit bias $\delta > 0$, since the true key's green list receives preferential sampling at every step. Full proofs are provided in Appendix B.

Theorem 3.2 guarantees that error probability decays exponentially with text length N, provided the watermark injects a positive bias ($\Delta_{min} > 0$). Even if the image decoder is noisy and produces a large candidate set, the text oracle will asymptotically converge to the correct key. It is important to note that Theorem 3.2 assumes the true key k^* is present in the candidate set $\mathcal{K}_{cand}$. The overall success probability of CoSignature is therefore the joint probability of two events:

$$P_{success} = P(\text{Oracle Correct} \mid k^* \in \mathcal{K}_{cand}) \cdot P(k^* \in \mathcal{K}_{cand})$$

Theorem 3.2 proves that the first term approaches 1 as text length N increases. The second term, $P(k^* \in \mathcal{K}_{cand})$, depends entirely on the robustness of the image channel.

The necessity of CRC filtering: This dependency highlights the necessity of the CRC filter in managing the candidate set size M. Recall that the error bound in Theorem 3.2 scales linearly with the number of candidates ($P_{error} \propto M$). Without a filter, increasing the OSD search depth k_{depth} to handle noisy images would exponentially increase M, thereby degrading the text oracle's reliability. The CRC filter breaks this trade-off: it allows us to increase k_{depth} (e.g., from 10 to 16) to boost the recall term $P(k^* \in \mathcal{K}_{cand})$, while simultaneously rejecting nearly all invalid permutations to keep M small (typically ≤ 5). This ensures that high robustness in the image channel does not come at the cost of accuracy in the text channel.

Table 2: Main Results on Clean Data. We evaluate performance across independent verification, binding verification, and swap detection. All TPRs are reported at fixed FPR=1.0%.

Metric	Janus-Pro	Qwen	Qwen3-VL	BAGEL+PS
Independent Verification				
Text TPR (System Key)	98.6%	96.7%	99.8%	98.8%
Image Exact Match	92.6%	97.3%	89.6%	76.2%
Binding Verification				
Binding TPR	**98.7%**	**97.4%**	**99.0%**	**96.7%**
Swap Detection Accuracy				
WM-WM Swap	92.1%	91.0%	99.0%	90.0%
NoWM(img)+WM Swap	98.7%	96.7%	99.8%	98.8%
WM+NoWM(txt) Swap	97.0%	99.0%	99.0%	99.0%
Avg Swap Detection	**95.9%**	**95.6%**	**99.3%**	**95.9%**

Finally, we provide security analysis in Appendix C.10.

4 Experiments

Dataset. Following the prompt setup of Jovanović et al. (2025), we generate 1000 interleaved text-and-image pairs using prompts derived from the ImageNet-1K class (Deng et al., 2009) (templates provided in Appendix C.2).

Model Architectures. To demonstrate universality, we evaluate CoSignature on four distinct model architectures: 1) *Native Unified* (Janus-Pro-7B (Chen et al., 2025)): An autoregressive transformer that generates image (384×384) and text tokens in a single shared stream; 2) *Composite Pipeline* (Qwen-Pipeline): A decoupled system combining Qwen3-8B (Yang et al., 2025) and Qwen-Image (1024×1024) (Wu et al., 2025), representing the current mainstream industry setup; 3) *Qwen3-VL Pipeline* (Qwen3-VL-8B): A vision-language model pipeline using Qwen3-VL-8B (**?**) for joint understanding and generation; 4) *BAGEL Pipeline* (BAGEL-7B-MoT (Deng et al., 2025)): A mixture-of-transformers model for multimodal generation.

Watermarking Configuration. For the text channel, we set a context width $h = 4$, and a green list ratio $\gamma = 0.25$. The logit bias δ is set to 2.0. For the image channel, we use VideoSeal (Fernandez et al., 2024) as the default image watermarker to embed a 256-bit payload, comprising a 224-bit Content ID and a 32-bit Cyclic Redundancy Check. To demonstrate modularity, we additionally evaluate with PixelSeal (Souček et al., 2025) as a drop-in replacement on the BAGEL pipeline. All models are evaluated with a fixed temperature $T = 0.7$ and Top-$p = 0.9$ sampling to ensure diversity. Experiments were conducted on a single NVIDIA H200 GPU.

4.1 Main Results

In this section, we evaluate the performance of CoSignature, using a fixed text generation length of 500 tokens, and the fixed search depth of $D = 16$.

Detectability Evaluation. We first evaluate the detection performance on undistorted generations. We report the TPR for three distinct detection regimes defined in Section 3.3: independent verification, binding verification and swap detection. For the latter, we further disaggregate performance across three mismatch scenarios: watermarked image and text swaps (WM-WM), unwatermarked image paired with watermarked text (NoWM(image)+WM), and watermarked image paired with unwatermarked text (WM+NoWM(text)). The results are summarized in Table 2.

As shown in Table 2, CoSignature maintains robust independent detectability across all four architectures, with text TPR of 96.7%–99.8% and image exact match of 76.2%–97.3%. The system key detection is mathematically equivalent to standard independent watermarking (same KGW algorithm, same bias δ), meaning CoSignature adds binding and swap detection *without sacrificing single-modality performance*. For binding verification, CoSignature achieves 96.7%–99.0% across all architectures. Under independent watermarking, swap detection is fundamentally impossible since both components pass their respective checks. In contrast, CoSignature achieves 95.6%–99.3% average swap detection, with the critical WM-WM scenario (valid but mismatched pairs) reaching 90.0%–99.0%.

Table 3: Robustness Results. Bit Error Rate (BER), Image-Only (Img) and CoSignature (CoS) content ID recovery accuracy (%), shown for the two extremes of the architecture range: Janus-Pro (native autoregressive, lowest resolution and hence highest BER) and Qwen3-VL (composite pipeline). Q denotes JPEG quality factor. All four pipelines and all ten distortions are reported in Tables 5 and 10, Appendix C.7.

	Janus-Pro			**Qwen3-VL**		
Attack	BER	Img	CoS	BER	Img	CoS
Resize (0.25×)	10.6	2.3	**24.0**	0.3	87.4	**99.2**
Crop (80%)	3.1	14.9	**73.2**	3.3	18.3	**70.6**
JPEG (Q=75)	3.7	14.7	**64.2**	0.6	75.5	**96.6**
JPEG (Q=50)	16.5	0.2	**8.1**	2.7	35.6	**78.4**
Blur (σ=2)	0.6	64.5	**97.3**	0.2	88.5	**98.8**
Noise (σ=10)	32.9	0.0	0.0	6.0	12.5	**51.5**

Quality Evaluation. CoSignature imposes negligible quality degradation: the Qwen pipeline sees only a marginal perplexity increase (2.88 → 3.15) and preserves CLIP alignment (24.20 vs. 24.27). Full quality results are in Appendix C.3.

Robustness Evaluation. We evaluate CoSignature's resilience against common image distortions. In all tables, Q denotes the JPEG quality factor. We report exact match accuracy (percentage of correctly recovered content IDs) comparing image-only decoding against CoSignature's joint decoder. Table 3 summarizes the two extremes of the architecture range; results for all four pipelines and all ten distortions are in Appendix C.7.

CoSignature significantly enhances recovery: under JPEG (Q=75) on Janus-Pro, accuracy improves from 14.7% to 64.2%; under crop 80% on Qwen3-VL, from 18.3% to 70.6%. The system corrects corrupted payloads provided bit errors are concentrated in the D least-confident positions. This creates a "Safe Operating Region" significantly wider than standard decoding, which fails at the first bit error. Notably, CoSignature exhibits a distinct "plateau" behavior: it maintains near-perfect recovery as noise increases, until the distortion overwhelms the search capacity or corrupts the CRC (e.g., Gaussian noise σ=10 on Janus with 32.9% BER). The robustness lift relies on CRC preservation; when noise overwhelms the CRC, recovery fails for both methods.

Sensitivity to Text Paraphrase. We evaluate resistance to LLM-based paraphrase attacks at three intensity levels (Table 7, Appendix C.4), all produced by Qwen3-VL-8B-Instruct in text-only mode at temperature 0.7 and differing only in how much change the instruction licenses. Because the watermark is carried by token identity, we characterize each level by the fraction of original tokens it retains: *synonym* retains a token Jaccard of 0.74, *moderate* 0.48, and *heavy* 0.35, the last at essentially unchanged length (ratio 1.01), confirming that heavy rewriting is a genuine rewrite rather than a truncation. CoSignature remains robust to synonym replacement and light paraphrasing, with performance declining under heavy rewriting as expected—significant semantic alterations disrupt the statistical green-list signal.

Graceful Degradation Under Combined Attack. Binding is the strongest claim CoSignature makes, and it is also the first to fail: it requires signal in *both* channels. We therefore ask what remains verifiable when binding is lost. Under the hardest combined attack we evaluate—heavy rewriting of the text together with JPEG (Q=75) on the image—binding is verified for 33.3% of pairs, but only 0.3% of pairs lose all verifiable provenance. The dominant outcome is not collapse but demotion: in 56.3% of cases the image still yields the exact Content ID, so the artifact still resolves to its registered record, and what is lost is specifically the evidence that *this* text accompanied it; a further 9.7% retain independent provenance in both modalities. Under a moderate rewrite with the same image attack, binding holds for 76.0% and no pair loses everything. The near-absence of the text-only outcome (0.3%) quantifies the channel asymmetry: under heavy rewriting the text channel fails well before the image channel. Full breakdown in Appendix C.4.

Verification Speed. Per-sample verification latency is reported in Table 13 (Appendix C.7). At search depth $D = 6$, verification takes ∼8ms per sample, suitable for online deployment. At $D = 16$, parallelized OSD completes in ∼438ms with 8-way parallelism, well within interactive latency for forensic-grade deep search.

4.2 Ablation Studies

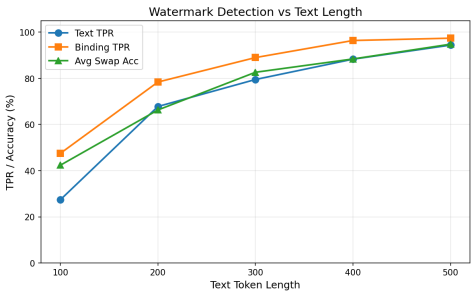


Figure 2: Impact of Text Length. Detection metrics across token counts (100–500) at FPR=1%.

Impact of Text Length. In Figure 2, we analyze the sensitivity of detection performance to generation length using the Qwen pipeline. As expected for statistical watermarking schemes, the true TPR is highly correlated with the token count, governed by the accumulation of the z-score signal. The results indicate that detection becomes reliable at paragraph-level lengths. At 300 tokens, both independent text TPR and binding TPR are $\approx 80\%$. As the length increases to 500 tokens, performance saturates near optimality, with the text TPR at 96.7% and binding TPR at 97.4%. This scaling behavior validates that CoSignature is well-suited for typical multimodal generation tasks, where responses naturally exceed 300 tokens.

Impact of the CRC Filter and Search Depth. We investigate the role of the cyclic redundancy check (CRC) as a validity filter using Janus-Pro model (since the lower resolution image has more bit errors to correct). We evaluate three configurations: a shallow search with and without CRC ($D = 6$), and deep search with CRC ($D = 16$). We omit the deep search without CRC setting as the exponential growth of the search space makes it computationally prohibitive and prone to false positives. Key results are summarized in Table 4, with the full set of results detailed in Table 9 in Appendix C.5. The results show that even at a fixed search depth ($D = 6$), enabling the CRC filter consistently improves recovery accuracy. By strictly rejecting candidates with invalid checksums, the CRC prevents the text oracle from converging on false bit sequences that statistically correlate with the text but are actually invalid. This empirical result also validates the theoretical bounds established in Theorem 3.2. Importantly, the CRC also allows deeper candidate searches without overwhelming the decoder. This deep search capability enables CoSignature to recover correct payloads that have drifted significantly from the centroid, boosting the recovery rate.

Table 4: Impact of the Search Depth (D) and CRC Filtering. We compare recovery rates on Janus-Pro under JPEG compression. "No CRC" relies solely on the text oracle. "With CRC" applies a validity filter, enabling deeper searches ($D = 16$).

Attack Type	**Baseline Img Only**	**Standard** ($D = 6$) **No CRC**	**Standard** ($D = 6$) **w/ CRC**	**Deep** ($D = 16$) **w/ CRC**
Clean	92.6%	94.9%	99.4%	**99.8%**
Blur ($\sigma = 2$)	64.5%	87.1%	92.5%	**97.3%**
Crop (80%)	14.9%	43.8%	48.3%	**73.2%**
JPEG (Q=75)	14.7%	40.4%	42.9%	**64.2%**

Furthermore, we show that our joint source-channel decoder matches or exceeds an idealized channel code given the same redundancy at a matched false-accept rate, and that the text channel substitutes for checksum redundancy directly: shrinking the checksum from 32 bits to 8 leaves a conventional decoder unable to identify a payload, while our decoder retains $\approx 98\%$ of the achievable recovery rate, returning 24 bits to payload capacity. We defer these experiments to Appendix C.8 and Appendix C.9.

5 Conclusion

We introduced CoSignature, a modular framework that transforms cross-modality binding into a robust cryptographic test, applicable to any image watermarker and sampling-based LLM watermarker. Empirically, CoSignature achieves >96% binding accuracy and >95% swap detection across four diverse architectures, with verification latency as low as 8ms. A key insight is that the text channel can substitute for checksum redundancy in the image payload: it performs with 8 check bits the disambiguation a 32-bit CRC performs, returning 24 bits to Content ID capacity, and at a matched false-accept rate the joint decoder is never worse than an idealized channel code given the same redundancy. While this work focused on image-text pairs, the principle extends naturally to audio-text or multi-modal triads, and CoSignature complements metadata standards like C2PA by providing persistent content-embedded binding. A current limitation is that CoSignature inherits the robustness profile of the underlying image watermarker; in particular, diffusion-based regeneration attacks remain a fundamental open problem for the field. As more robust watermarkers emerge, CoSignature will automatically benefit due to its modular design.

REFERENCES

Scott Aaronson. Watermarking of large language models. https://www.scottaaronson.com/talks/watermark.ppt, 2022.

Ross Anderson. *Security Engineering: A Guide to Building Dependable Distributed Systems*, chapter 24, pp. 815–863. John Wiley & Sons, Ltd, 2020. ISBN 9781119644682. doi: https://doi.org/10.1002/9781119644682.ch24.

C2PA. Coalition for Content Provenance and Authenticity (C2PA). URL https://c2pa.org/.

Yapei Chang, Kalpesh Krishna, Amir Houmansadr, John Frederick Wieting, and Mohit Iyyer. Postmark: A robust blackbox watermark for large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8969–8987, 2024.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025.

Miranda Christ, Sam Gunn, and Or Zamir. Undetectable watermarks for language models. In *The Thirty Seventh Annual Conference on Learning Theory*, pp. 1125–1139. PMLR, 2024.

Sumanth Dathathri, Abigail See, Sumedh Ghaisas, Po-Sen Huang, Rob McAdam, Johannes Welbl, Vandana Bachani, Alex Kaskasoli, Robert Stanforth, Tatiana Matejovicova, et al. Scalable watermarking for identifying large language model outputs. *Nature*, 634(8035):818–823, 2024.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. IEEE, 2009.

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.

Yuying Ge, Sijie Zhao, Jinguo Zhu, Yixiao Ge, Kun Yi, Lin Song, Chen Li, Xiaohan Ding, and Ying Shan. Seed-x: Multimodal models with unified multi-granularity comprehension and generation. *arXiv preprint arXiv:2404.14396*, 2024.

Sven Gowal, Rudy Bunel, Florian Stimberg, David Stutz, Guillermo Ortiz-Jimenez, Christina Kouridi, Mel Vecerik, Jamie Hayes, Sylvestre-Alvise Rebuffi, Paul Bernard, et al. Synthid-image: Image watermarking at internet scale. *arXiv preprint arXiv:2510.09263*, 2025.

Abe Hou, Jingyu Zhang, Tianxing He, Yichen Wang, Yung-Sung Chuang, Hongwei Wang, Lingfeng Shen, Benjamin Van Durme, Daniel Khashabi, and Yulia Tsvetkov. Semstamp: A semantic watermark with paraphrastic robustness for text generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 4067–4082, 2024a.

Abe Hou, Jingyu Zhang, Yichen Wang, Daniel Khashabi, and Tianxing He. k-semstamp: A clustering-based semantic watermark for detection of machine-generated text. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 1706–1715, 2024b.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Nikola Jovanović, Ismail Labiad, Tomáš Souček, Martin Vechev, and Pierre Fernandez. Watermarking autoregressive image generation. *arXiv preprint arXiv:2506.16349*, 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*, pp. 17061–17084. PMLR, 2023a.

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. *arXiv preprint arXiv:2306.04634*, 2023b.

Rohith Kuditipudi, John Thickstun, Tatsunori Hashimoto, and Percy Liang. Robust distortion-free watermarks for language models. *arXiv preprint arXiv:2307.15593*, 2023.

Aiwei Liu, Leyi Pan, Xuming Hu, Shiao Meng, and Lijie Wen. A semantic invariant robust watermark for large language models. *arXiv preprint arXiv:2310.06356*, 2023.

Shuliang Liu, Zheng Qi, Jesse Jiaxi Xu, Yibo Yan, Junyan Zhang, He Geng, Aiwei Liu, Peijie Jiang, Jia Liu, Yik-Cheung Tam, et al. Vla-mark: A cross modal watermark for large vision-language alignment models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 26420–26438, 2025a.

Zhiheng Liu, Weiming Ren, Haozhe Liu, Zijian Zhou, Shoufa Chen, Haonan Qiu, Xiaoke Huang, Zhaochong An, Fanny Yang, Aditya Patel, et al. Tuna: Taming unified visual representations for native unified multimodal models. *arXiv preprint arXiv:2512.02014*, 2025b.

Aleksandar Petrov, Pierre Fernandez, Tomáš Souček, and Hady Elsahar. We can hide more bits: The unused watermarking capacity in theory and in practice. *arXiv preprint arXiv:2510.12812*, 2025.

Wenjie Qu, Dong Yin, Zixin He, Wei Zou, Tianyang Tao, Jinyuan Jia, and Jiaheng Zhang. Provably robust multi-bit watermarking for ai-generated text via error correction code. *arXiv preprint arXiv:2401.16820*, 2024.

Jie Ren, Han Xu, Yiding Liu, Yingqian Cui, Shuaiqiang Wang, Dawei Yin, and Jiliang Tang. A robust semantics-based watermark for large language model against paraphrasing. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 613–625, 2024.

Tom Sander, Pierre Fernandez, Alain Durmus, Teddy Furon, and Matthijs Douze. Watermark anything with localized messages. *arXiv preprint arXiv:2411.07231*, 2024.

Zhouxing Shi, Yihan Wang, Fan Yin, Xiangning Chen, Kai-Wei Chang, and Cho-Jui Hsieh. Red teaming language model detectors with language models. *Transactions of the Association for Computational Linguistics*, 12:174–189, 2024.

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.

Guo-Hua Wang, Shanshan Zhao, Xinjie Zhang, Liangfu Cao, Pengxin Zhan, Lunhao Duan, Shiyin Lu, Minghao Fu, Jianshan Zhao, Yang Li, and Qing-Guo Chen. Ovis-u1 technical report. *arXiv preprint arXiv:2506.23044*, 2025.

Lean Wang, Wenkai Yang, Deli Chen, Hao Zhou, Yankai Lin, Fandong Meng, Jie Zhou, and Xu Sun. Towards codable watermarking for injecting multi-bits information to llms. In *Proceedings of the International Conference on Learning Representations (ICLR2024)*, 2024.

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.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025.

Yihan Wu, Zhengmian Hu, Hongyang Zhang, and Heng Huang. Dipmark: A stealthy, efficient and resilient watermark for large language models. In *International Conference on Machine Learning*, 2024.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Ning Yu, Vladislav Skripniuk, Dingfan Chen, Larry Davis, and Mario Fritz. Responsible disclosure of generative models using scalable fingerprinting. *arXiv preprint arXiv:2012.08726*, 2020.

Lijun Zhang, Xiao Liu, Antoni Viros i Martin, Cindy Xiong Bearfield, Yuriy Brun, and Hui Guan. Attack-resilient image watermarking using stable diffusion. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024a. URL https://openreview.net/forum?id=e6KrSouGHJ.

Xinjie Zhang, Jintao Guo, Shanshan Zhao, Minghao Fu, Lunhao Duan, Jiakui Hu, Yong Xien Chng, Guo-Hua Wang, Qing-Guo Chen, Zhao Xu, et al. Unified multimodal understanding and generation models: Advances, challenges, and opportunities. *arXiv preprint arXiv:2505.02567*, 2025.

Xuanyu Zhang, Youmin Xu, Runyi Li, Jiwen Yu, Weiqi Li, Zhipei Xu, and Jian Zhang. V2a-mark: Versatile deep visual-audio watermarking for manipulation localization and copyright protection. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 9818–9827, 2024b.

Xuandong Zhao, Prabhanjan Ananth, Lei Li, and Yu-Xiang Wang. Provable robust watermarking for ai-generated text. *arXiv preprint arXiv:2306.17439*, 2023.

Xuandong Zhao, Sam Gunn, Miranda Christ, Jaiden Fairoze, Andres Fabrega, Nicholas Carlini, Sanjam Garg, Sanghyun Hong, Milad Nasr, Florian Tramer, et al. SoK: Watermarking for AI-generated content. *arXiv preprint arXiv:2411.18479*, 2024a.

Xuandong Zhao, Lei Li, and Yu-Xiang Wang. Permute-and-flip: An optimally robust and watermarkable decoder for llms. *arXiv preprint arXiv:2402.05864*, 2024b.

A THE SCORE STATISTICS

We derive the theoretical mean μ_w and standard deviation σ_w of the weighted scoring function $E(k)$ under the Null Hypothesis (H_0).

Definitions. Under the Null Hypothesis, we assume the candidate content key k is uncorrelated with the generated text. However, since the text is generated with the system watermark active, tokens are biased toward the system green list $\mathcal{G}_{sys}$. We condition on the observed system green token fraction $\hat{p}_{sys} = N_{sys}/n$, where N_{sys} is the number of tokens falling in $\mathcal{G}_{sys}$. For any given token x_t:

- Event A: $x_t \in \mathcal{G}_{sys}$, with observed probability $\hat{p}_{sys}$ (elevated above γ due to system watermark).
- Event B: $x_t \in \mathcal{G}_{cont}(k)$, with $P(B) = \gamma$ (independent of system key under H_0).

Since the content key seed is independent of the system key, $P(A \cap B) = \hat{p}_{sys} \cdot \gamma$.

Random Variable Construction. The scoring function assigns weights based on the partition:

- **Double-Green** ($A \cap B$): Weight $W = 2$. Probability $p_2 = \hat{p}_{sys} \cdot \gamma$.
- **Single-Green** ($B \setminus A$): Weight $W = 1$. Probability $p_1 = (1 - \hat{p}_{sys}) \cdot \gamma$.
- **Red/Sys-Only**: Weight $W = 0$. Probability $p_0 = 1 - p_1 - p_2$.

A.1 DERIVATION OF THE MEAN (μ_w)

The expected score for a single token, $\mathbb{E}[S_t]$, is:

$$\begin{aligned}\mathbb{E}[S_t] &= 2 \cdot p_2 + 1 \cdot p_1 + 0 \cdot p_0 \\ &= 2(\hat{p}_{sys} \cdot \gamma) + 1((1 - \hat{p}_{sys}) \cdot \gamma) \\ &= 2\hat{p}_{sys}\gamma + \gamma - \hat{p}_{sys}\gamma \\ &= \gamma(1 + \hat{p}_{sys})\end{aligned}$$

For a sequence of length n, the total expected score is:

$$\mu_w = n \cdot \mathbb{E}[S_t] = n\gamma(1 + \hat{p}_{sys}) \tag{6}$$

Note: when the text is unwatermarked ($\hat{p}_{sys} = \gamma$), this reduces to the unconditional formula $\mu_w = n\gamma(1 + \gamma)$.

A.2 DERIVATION OF THE VARIANCE (σ_w^2)

Because we *condition* on the observed system-green fraction, membership of x_t in $\mathcal{G}_{sys}$ is a deterministic property of the given text, not a random event; the only randomness under H_0 is membership in $\mathcal{G}_{cont}(k)$, which occurs with probability γ independently of the system key. It is therefore cleanest to split the sequence by the value of the indicator $a_t = \mathbb{I}[x_t \in \mathcal{G}_{sys}]$, of which exactly $N_{sys} = n\hat{p}_{sys}$ are equal to 1.

For a token with $a_t = 1$ the score is $S_t = 2 \cdot \mathbb{I}[x_t \in \mathcal{G}_{cont}(k)]$, so $\text{Var}(S_t) = 4\gamma(1 - \gamma)$. For a token with $a_t = 0$ the score is $S_t = \mathbb{I}[x_t \in \mathcal{G}_{cont}(k)]$, so $\text{Var}(S_t) = \gamma(1 - \gamma)$. Assuming independence across tokens (the standard approximation for H_0, since the green lists at distinct positions are seeded by distinct contexts), the variances add:

$$\begin{aligned}\sigma_w^2 &= n\hat{p}_{sys} \cdot 4\gamma(1 - \gamma) + n(1 - \hat{p}_{sys}) \cdot \gamma(1 - \gamma) \\ &= n\gamma(1 - \gamma)\big(4\hat{p}_{sys} + 1 - \hat{p}_{sys}\big) \\ &= n\gamma(1 - \gamma)(1 + 3\hat{p}_{sys})\end{aligned}$$

giving the standard deviation:

$$\sigma_w = \sqrt{n\gamma(1 - \gamma)(1 + 3\hat{p}_{sys})} \tag{7}$$

Two remarks. First, the conditional variance is strictly smaller than the variance one would obtain by treating $\mathcal{G}_{sys}$ membership as an additional Bernoulli($\hat{p}_{sys}$) event, namely $n[\gamma(1 + 3\hat{p}_{sys}) - \gamma^2(1 + \hat{p}_{sys})^2]$; the difference is exactly the variance contributed by fluctuation in N_{sys}, which conditioning removes. Using the unconditioned expression with a conditional mean would overstate σ_w and lose power. Second, the conditioning matters precisely because $\hat{p}_{sys} > \gamma$ for system-watermarked text: substituting γ for $\hat{p}_{sys}$ in μ_w leaves a systematic positive drift in z for *every* candidate key, which is the failure mode exploited by the forgery attack in Section 4 and eliminated by Eq. 5.

B PROOF FOR THEOREM 3.2 AND THE COROLLARY

Theorem B.1 (Theorem 3.2 restated)**.** *Let $T = (x_1, \ldots, x_N)$ be the generated text sequence. Let $\mathcal{K}_{cand}$ be a set of M candidate keys containing the true content key k^*. Let $X_t^{(k)}$ denote the weighted watermark score for token x_t under key k, and let $Y_t = X_t^{(k^*)} - X_t^{(k')}$ be the score difference at step t. Assume there exists a deterministic constant $\Delta_{min} > 0$ such that:*

$$\mathbb{E}[Y_t \mid H_{t-1}] \geq \Delta_{min}, \quad \textit{almost surely for all } t. \tag{8}$$

This condition holds whenever the watermark injects a positive logit bias $\delta > 0$, since the true key's green list receives preferential sampling at every generation step. The probability of detection error P_{error} is bounded by:

$$P_{error} \leq (M-1) \cdot \exp\left(-\frac{N \cdot \Delta_{min}^2}{8}\right). \tag{9}$$

Proof. The probability of detection error P_{error} is defined as the event where any false candidate $k' \in \mathcal{K}_{cand} \setminus \{k^*\}$ achieves a higher Weighted Z-score than k^*.

Let the score for a single token t under a candidate key k be the random variable $X_t^{(k)}$, defined as

$$X_t^{(k)} = 2 \cdot \mathbb{I}[x_t \in \mathcal{G}_{sys} \cap \mathcal{G}_{cont}(k)] + 1 \cdot \mathbb{I}[x_t \in \mathcal{G}_{cont}(k) \setminus \mathcal{G}_{sys}]. \tag{10}$$

Let k^* be the true key and consider a single false candidate k'. Let $Y_t = X_t^{(k^*)} - X_t^{(k')}$ be the score difference at step t. An error occurs between this pair if $\sum Y_t \leq 0$.

Since the text generation is autoregressive, the distribution of Y_t depends on the sequence of tokens generated prior to step t, denoted as $H_{t-1} = \{x_1, ..., x_{t-1}\}$. We construct a Doob Martingale sequence Z_m. Let $S_m = \sum_{t=1}^{m} Y_t$ be the partial sum. We define

$$Z_m = S_m - \sum_{t=1}^{m} \mathbb{E}[Y_t \mid H_{t-1}]. \tag{11}$$

By definition, Z_m is a martingale because $\mathbb{E}[Z_m \mid H_{m-1}] = Z_{m-1}$.

Let $D_t = Z_t - Z_{t-1} = Y_t - \mathbb{E}[Y_t \mid H_{t-1}]$ be the martingale differences. Since the raw score difference Y_t is strictly bounded in $[-2, 2]$, the centered variable D_t takes values in an interval of length at most $c_t = 4$.

We seek the probability that the actual sum S_N drops below 0. By our assumption, $\sum_{t=1}^{N} \mathbb{E}[Y_t \mid H_{t-1}] \geq N\Delta_{min}$ almost surely. Therefore:

$$P(S_N \leq 0) \leq P\left(Z_N \leq -N\Delta_{min}\right) \tag{12}$$

Since the martingale differences are bounded, we apply Azuma's Inequality with the *deterministic* threshold $\epsilon = N\Delta_{min}$. For a martingale with differences D_t taking values in intervals of length c_t:

$$P(Z_N \leq -\epsilon) \leq \exp\left(-\frac{2\epsilon^2}{\sum_{t=1}^{N} c_t^2}\right). \tag{13}$$

Substituting $\epsilon = N\Delta_{min}$ and $\sum c_t^2 = 16N$, we have

$$P(S_N \leq 0) \leq \exp\left(-\frac{2(N\Delta_{min})^2}{16N}\right) = \exp\left(-\frac{N\Delta_{min}^2}{8}\right). \tag{14}$$

The total detection error is the probability that *at least one* of the $M-1$ false candidates exceeds the true key. By the Union Bound, we have

$$P_{error} = P\left(\bigcup_{k' \neq k^*} \{Z(k') \geq Z(k^*)\}\right) \leq \sum_{k' \neq k^*} P(Z(k') \geq Z(k^*)). \tag{15}$$

Substituting the pairwise bound, we have

$$P_{error} \leq (M-1) \cdot \exp\left(-\frac{N\Delta_{min}^2}{8}\right). \tag{16}$$

Since M is small (controlled by the CRC filter), the exponential decay in N dominates the error probability, guaranteeing convergence. □

The following corollary provides the minimum verified length of the generated text to ensure the binding detection accuracy.

Corollary B.2 (Minimum Verified Length)**.** *To ensure the binding error probability remains below a target threshold η, the generated text length N must satisfy the lower bound:*

$$N_{min} \geq \frac{8}{\Delta_{min}^2} \ln\left(\frac{M-1}{\eta}\right) \tag{17}$$

where M is the size of the candidate set and Δ_{min} is the minimum per-step signal margin.

Corollary B.2 dictates that the required text length scales quadratically with the inverse of the watermark strength Δ_{min} and logarithmically with the candidate search space M.

C Additional Experiments and Details

C.1 Generation Example



Image
Detection: Watermark Detected
Payload: `Content_ID: 8A2F...`

Prompt: "I am wondering about the Golden Retriever. Can you teach me something about it in a way that is easy to understand and also generate a picture of it?"

What is a Golden Retriever?
A Golden Retriever is a large, friendly dog breed known for its soft golden coat, kind eyes, and playful nature. They're like the "best friend" of dogs!
Why Are They Popular?
-Friendly and Loving: They love people and get along well with kids, other dogs, and even strangers.
-Smart and trainable: They're quick learners and excel in tasks like obedience training.
-Photogenic: Their shiny, golden fur makes them look like they're always smiling!
What Do They Look Like?
-Size: Medium to large (think "dog-sized teddy bear").
-Coat: Thick, wavy, and golden (like sunlight on a lake).
-Eyes: Bright, kind, and usually brown.
-Tail: Bushy and waggy (they're always happy!)...

Text Detection: System Signal ($z = 6.4$) **Binding:** Match ($S_{binding} > \tau_{bind}$)

Figure 3: A sample generation produced by the Qwen pipeline. Despite the strong binding signal embedded in both modalities, the visual quality remains high and the text retains its structure, fluency, and helpful tone.

C.2 Prompt Templates

To ensure semantic alignment between the text and image, we use the following template for all ImageNet classes:

> *"I am wondering about the following: {class_label}. Can you teach me something about it in a way that is easy to understand and also generate a picture of it?"*

This prompt structure forces the model to treat the image generation as a sub-task of the textual explanation, creating the "binding" context we aim to protect.

C.3 Quality Evaluation

Next, we evaluate the impact of CoSignature on generation quality. Our primary metrics are text perplexity (measuring fluency), and the CLIP score (measuring semantic alignment). We omit a comparative evaluation of image quality, as it is determined by the underlying image watermarking model, which is orthogonal to our cross-modality scheme. We use Qwen3-32B model to evaluate the perplexity of the generated text. Using a larger model ensures that the metric reflects true naturalness. We report the results for both unwatermarked and watermarked generations.

As shown in Table 6, the perplexity degradation is minimal. For the Qwen pipeline, the perplexity increases slightly from 2.88 to 3.15. Interestingly, for Janus-Pro generation results, the mean

Table 5: **Comprehensive Robustness Evaluation (Janus-Pro-7B & Qwen-Pipeline).** Image-Only vs. CoSignature content ID recovery rate under diverse distortions.

	Native Unified (Janus-Pro-7B)				Composite Pipeline (Qwen3-8B + Qwen-Image)			
Attack Type	**Bit Err (BER) ↓**	**Img Only Acc**	**CoSignature (Ours)**	**Lift (Δ)**	**Bit Err (BER) ↓**	**Img Only Acc**	**CoSignature (Ours)**	**Lift (Δ)**
Geometric Distortions								
Resize (0.5×)	0.2%	84.2%	**99.0%**	+14.8%	0.2%	89.3%	**99.5%**	+10.2%
Resize (0.25×)	10.6%	2.3%	**24.0%**	+21.7%	0.2%	87.0%	**99.3%**	+12.3%
Crop (90%)	0.8%	56.6%	**95.6%**	+39.0%	0.7%	62.7%	**96.1%**	+33.4%
Crop (80%)	3.1%	14.9%	**73.2%**	+58.3%	3.2%	19.5%	**71.2%**	+51.7%
Compression & Noise								
JPEG ($Q = 95$)	0.3%	83.1%	**99.3%**	+16.2%	0.2%	87.2%	**99.5%**	+12.3%
JPEG ($Q = 85$)	1.0%	52.6%	**94.7%**	+42.1%	0.3%	82.9%	**99.1%**	+16.2%
JPEG ($Q = 75$)	3.7%	14.7%	**64.2%**	+49.5%	0.5%	74.5%	**97.1%**	+22.6%
JPEG ($Q = 50$)	16.5%	0.2%	**8.1%**	+7.9%	2.4%	37.7%	**78.6%**	+40.9%
Gaussian Blur ($\sigma = 2$)	0.6%	64.5%	**97.3%**	+32.8%	0.2%	88.6%	**99.1%**	+10.5%
Gaussian Blur ($\sigma = 4$)	66.2%	0.0%	0.0%	+0.0%	0.3%	81.1%	**98.8%**	+17.7%
Gaussian Noise ($\sigma = 10$)	32.9%	0.0%	0.0%	+0.0%	6.1%	11.5%	**50.4%**	+38.9%
Photometric Shifts								
Brightness (+30%)	2.8%	42.5%	**78.4%**	+35.9%	7.4%	50.9%	**71.6%**	+20.7%
Brightness (−30%)	0.2%	86.5%	**98.8%**	+12.3%	0.2%	85.7%	**98.9%**	+13.2%
Contrast (+30%)	0.8%	63.9%	**94.0%**	+30.1%	0.5%	80.7%	**97.7%**	+17.0%
Contrast (−30%)	0.1%	91.6%	**99.5%**	+7.9%	0.2%	89.8%	**99.1%**	+9.3%

Table 6: **Generation Quality Assessment.** We compare the text fluency (Perplexity using Qwen3-32B) and semantic alignment (CLIP Score using openai/clip-vit-large-patch14) of clean versus watermarked generations. Lower PPL and higher CLIP indicate better quality.

Model	**Text Fluency (PPL) ↓**		**Image-Text Alignment (CLIP) ↑**	
	Clean	**Watermarked**	**Clean**	**Watermarked**
Qwen Pipeline	2.88	3.15	24.20	24.27
Janus-Pro-7B	6.44	6.29	26.60	23.50
Qwen3-VL-8B	3.65	4.60	23.01	22.54

perplexity actually decreases slightly (6.44 to 6.29) under watermarking. This also indicates that Janus-Pro text generation has a higher entropy (thus higher detection accuracy), and the small perturbation from the watermark does not degrade the text quality. Regarding semantic alignment, the CLIP scores show that the alignment is fully preserved (24.20 vs. 24.27) for the Qwen pipeline, while we observe a moderate drop in CLIP score (26.60 → 23.50) for Janus-Pro. This suggests that in a unified autoregressive model, the perturbation of text tokens may slightly decouple the semantic link with the preceding visual tokens. However, the score of 23.50 remains well within the range of high-quality image-text pairs, confirming that the system maintains strong semantic coherence.

C.4 Text Paraphrasing Attacks

To simulate varying degrees of semantic textual attacks, we utilized the Qwen3-8B as an adversary. We employed the following system prompts to control the attack intensity:

- Synonym replacement: "Only replace a few words with synonyms while keeping the sentence almost the same. Output only the modified text."
- Moderate paraphrasing: "Lightly paraphrase the following text while preserving the sentence structure. Output only the paraphrased text."
- Heavy rewriting: "Completely rewrite the following text using different words and sentence structures while preserving the core meaning. Output only the rewritten text."

Because the watermark signal is carried by token identity rather than meaning, we characterize each intensity by how much of the original token sequence it preserves. Over 300 articles, the mean token Jaccard against the original is 0.740 (synonym), 0.483 (moderate) and 0.345 (heavy), with mean length ratios of 0.91, 0.92 and 1.01 respectively. Heavy rewriting therefore discards roughly two thirds of the original tokens while leaving length essentially unchanged, confirming it is a genuine

rewrite rather than a truncation. These statistics let a reader locate a realistic attacker's operating point: binding survives while roughly half the tokens survive, and degrades sharply below that.

The results are presented in Table 7.

Table 7: **Robustness to Text Paraphrasing.** We evaluate detection rates (Text TPR and Binding TPR) under varying levels of LLM-based paraphrasing attacks across all four pipelines.

Attack Level	Janus-Pro-7B		Qwen-Pipeline		Qwen3-VL		BAGEL+PixelSeal	
	Text	Bind	Text	Bind	Text	Bind	Text	Bind
Original (Clean)	98.6%	98.7%	96.7%	97.9%	99.8%	99.9%	98.8%	99.1%
Synonym Swap	84.9%	80.5%	96.3%	96.9%	99.5%	99.6%	98.6%	98.7%
Light Paraphrase	80.6%	76.7%	96.2%	96.4%	99.8%	99.9%	98.3%	98.4%
Heavy Rewrite	79.7%	76.2%	60.2%	63.8%	80.9%	85.6%	62.9%	65.8%

Decomposing the Combined Attack. Table 7 reports whether binding survives, but not what remains when it does not. Since binding requires signal in both channels, it is by construction the first guarantee to fail, and the operationally relevant question is whether a failure is a graceful demotion or a total loss. We therefore assign every pair to exactly one of five mutually exclusive outcomes and evaluate the combination of a text rewrite with an image distortion (Qwen3-VL, n=300, Table 8).

Table 8: **Graceful degradation under combined attack.** Outcome distribution (%) on Qwen3-VL, n=300. *Bound* = binding verified; *Both indep.* = binding lost but the image ID is recovered and the text passes source detection; *Image only* / *Text only* = one modality remains independently verifiable; *All lost* = no verifiable provenance. The final column is the fraction retaining some verifiable provenance.

Attack	Bound	Both indep.	Image only	Text only	All lost	Any
Clean (no attack)	**100.0**	0.0	0.0	0.0	0.0	100.0
Moderate rewrite + JPEG (Q=75)	**76.0**	7.0	17.0	0.0	**0.0**	100.0
Heavy rewrite + JPEG (Q=75)	**33.3**	9.7	56.3	0.3	**0.3**	99.7

Two observations. First, degradation is graceful in a measurable sense: even under the hardest combination only 0.3% of pairs lose all verifiable provenance, and the dominant outcome is that the image still yields the exact Content ID (56.3%), so the artifact still resolves to its registered record while the evidence tying it to that specific text is lost. Second, the near-empty *Text only* column quantifies the channel asymmetry: under heavy rewriting the text channel fails well before the image channel, so binding failures are almost always attributable to the text side. This is a property inherited from token-level text watermarking rather than from the binding construction, and a paraphrase-robust text watermarker plugged into the same interface would be expected to shift this column directly.

C.5 Ablation: Impact of the CRC Filter and Search Depth

We provide the full results in Table 9.

C.6 Extension to Gumbel-Based Watermarking

While our current implementation instantiates the dual-key superposition using green-list biasing Kirchenbauer et al. (2023a), the principle generalizes to other sampling-based text watermarking schemes. Here we discuss how CoSignature can be extended to Gumbel sampling Aaronson (2022).

With Gumbel sampling, we generate two Gumbel noise vectors:

$$g_{sys} = \text{Gumbel}(\text{Hash}(k_{sys}, x_{<t})), \quad g_{sess} = \text{Gumbel}(\text{Hash}(k_{cont}, x_{<t})). \tag{18}$$

Two natural superposition strategies exist:

Additive superposition: Sample $x_t = \text{argmax}(\log p + g_{sys} + g_{sess})$. Both keys boost every token, but the combined noise is no longer Gumbel-distributed, introducing a small distortion to the output distribution.

Table 9: **Detailed Ablation: Impact of CRC Filtering and Search Depth.** We compare CoSignature's recovery rate across all evaluated distortions on Janus-Pro model. The addition of the CRC filter consistently improves precision at standard depth ($D = 6$) by rejecting invalid candidates. Crucially, the CRC filter enables the use of deep search ($D = 16$), which unlocks significant robustness gains for severe distortions (e.g., Crop 80% and JPEG Q=75) that are otherwise unrecoverable.

Distortion Type	**Baseline**	**Standard Search** ($D = 6$)		**Deep Search** ($D = 16$)
	Img Only	**No CRC**	**With CRC**	**With CRC**
No Attack (Clean)	92.6%	94.9%	99.4%	**99.8%**
Geometric Distortions				
Resize ($0.5\times$)	84.2%	93.9%	97.7%	**99.0%**
Resize ($0.25\times$)	2.3%	10.0%	12.1%	**24.0%**
Crop (90%)	56.6%	84.0%	90.3%	**95.6%**
Crop (80%)	14.9%	43.8%	48.3%	**73.2%**
Photometric & Noise				
Gaussian Blur ($\sigma = 2$)	64.5%	87.1%	92.5%	**97.3%**
Brightness ($+30\%$)	42.5%	65.6%	68.5%	**78.4%**
Brightness (-30%)	86.5%	94.0%	98.1%	**98.8%**
Contrast ($+30\%$)	63.9%	84.3%	88.3%	**94.0%**
Contrast (-30%)	91.6%	94.7%	99.0%	**99.5%**
JPEG Compression				
JPEG ($Q = 95$)	83.1%	94.1%	97.8%	**99.3%**
JPEG ($Q = 85$)	52.6%	84.2%	87.6%	**94.7%**
JPEG ($Q = 75$)	14.7%	40.4%	42.9%	**64.2%**
JPEG ($Q = 50$)	0.2%	1.7%	2.4%	**8.1%**

Element-wise maximum: Sample $x_t = \text{argmax}(\log p + \max(g_{sys}, g_{sess}))$. This preserves the Gumbel distribution via max-stability ($\max$ of two Gumbel$(0, 1) \sim$ Gumbel$(\ln 2, 1)$) and remains distortion-free. The trade-off is a reduction in signal strength for each individual key, since the maximum operation dilutes the per-key signal.

The optimal choice depends on the specific trade-off between detection power and text quality, requiring case-by-case analysis. The scope of our current work is introducing the general dual-key superposition principle; we leave the exploration of the optimal key superposition strategy for each sampling-based scheme to future work.

C.7 Additional Pipelines, Modularity, and Verification Speed

We report the full robustness evaluation for Qwen3-VL-8B and BAGEL-7B-MoT, the image watermarker modularity study, and verification speed benchmarks.

Modularity Along the Text Axis: a Gumbel Instantiation. Table 11 varies the image watermarker. To test the other half of the modularity claim we replace the *text* watermarker, substituting Gumbel-softmax sampling (Aaronson, 2022) for KGW while holding the image watermarker, model, resolution and joint decoder fixed. Only the sampling rule and the corresponding detection statistic change, so this isolates a single variable. Thresholds are calibrated at FPR=1% on non-watermarked text as elsewhere; n=300 with 95% Wilson intervals.

The robustness lift transfers as well: across the same eleven distortions the Gumbel instantiation's lift over image-only decoding tracks the KGW lift closely and differs in *both* directions (e.g. $+47.7$ vs. $+42.8$ on JPEG Q=50, $+49.3$ vs. $+52.3$ on crop 80%), as expected from the smaller subsample. Because the image-only column is identical by construction, this isolates exactly one variable and indicates that the joint source–channel decoding mechanism is not specific to a green-list watermark. The clean-data gap in binding (95.3 vs. 99.0) reflects the per-key signal dilution of the distortion-free max-superposition rather than a degradation of binding.

Verification Speed. We report per-sample verification timing in Table 13. The verification pipeline consists of two stages: (1) image watermark extraction on GPU, and (2) OSD candidate enumeration with CRC checking on CPU. The image extraction step is constant at 6.2ms regardless of search depth, as it simply runs the watermark detector forward pass.

At search depth $D = 6$, the OSD enumerates $2^6 = 64$ candidate bit patterns and checks each against the 32-bit CRC. This completes in 1.8ms, yielding a total verification time of $\sim$8ms—suitable for

Table 10: **Comprehensive Robustness Evaluation (Qwen3-VL-8B & BAGEL+PixelSeal).** Image-Only vs. CoSignature content ID recovery rate under diverse distortions.

	Qwen3-VL-8B Pipeline				BAGEL-7B-MoT + PixelSeal			
Attack Type	**Bit Err (BER) ↓**	**Img Only Acc**	**CoSignature (Ours)**	**Lift (Δ)**	**Bit Err (BER) ↓**	**Img Only Acc**	**CoSignature (Ours)**	**Lift (Δ)**
No Attack (Clean)	0.2%	89.6%	**99.2%**	+9.6%	0.3%	76.2%	**99.3%**	+23.1%
Geometric Distortions								
Resize (0.5×)	0.2%	89.6%	**99.4%**	+9.8%	0.4%	73.4%	**98.9%**	+25.5%
Resize (0.25×)	0.3%	87.4%	**99.2%**	+11.8%	0.5%	66.7%	**98.3%**	+31.6%
Crop (90%)	0.8%	62.8%	**96.0%**	+33.2%	0.8%	55.4%	**95.3%**	+39.9%
Crop (80%)	3.3%	18.3%	**70.6%**	+52.3%	0.9%	50.9%	**95.7%**	+44.8%
Compression & Noise								
JPEG ($Q = 95$)	0.3%	86.5%	**98.9%**	+12.4%	0.7%	59.7%	**97.2%**	+37.5%
JPEG ($Q = 85$)	0.4%	82.3%	**98.1%**	+15.8%	1.1%	44.2%	**93.3%**	+49.1%
JPEG ($Q = 75$)	0.6%	75.5%	**96.6%**	+21.1%	2.0%	25.8%	**81.5%**	+55.7%
JPEG ($Q = 50$)	2.7%	35.6%	**78.4%**	+42.8%	11.4%	2.3%	**16.2%**	+13.9%
Gaussian Blur ($\sigma = 2$)	0.2%	88.5%	**98.8%**	+10.3%	0.4%	71.9%	**99.1%**	+27.2%
Gaussian Blur ($\sigma = 4$)	0.4%	82.9%	**98.3%**	+15.4%	1.2%	40.0%	**91.2%**	+51.2%
Gaussian Noise ($\sigma = 10$)	6.0%	12.5%	**51.5%**	+39.0%	20.8%	0.0%	**1.9%**	+1.9%
Photometric Shifts								
Brightness (+30%)	9.7%	49.4%	**69.3%**	+19.9%	12.6%	30.0%	**55.2%**	+25.2%
Brightness (−30%)	0.3%	85.8%	**98.6%**	+12.8%	0.7%	62.6%	**96.9%**	+34.3%
Contrast (+30%)	0.6%	82.1%	**97.2%**	+15.1%	1.2%	49.1%	**91.6%**	+42.5%
Contrast (−30%)	0.2%	89.4%	**99.1%**	+9.7%	0.7%	67.7%	**96.8%**	+29.1%

Table 11: **Qwen Pipeline: VideoSeal vs. PixelSeal.** Img% = image-only accuracy; Orc% = CoSignature accuracy. CoSignature lifts both watermarkers.

Attack	**VS Img%**	**VS Orc%**	**PS Img%**	**PS Orc%**
Resize (0.25×)	87.0	99.3	89.2	99.5
Crop (80%)	19.5	71.2	70.9	97.6
JPEG (Q=75)	74.5	97.1	70.3	98.9
JPEG (Q=50)	37.7	78.6	20.8	64.7
Blur ($\sigma = 2$)	88.6	99.1	92.2	99.6
Noise ($\sigma = 10$)	11.5	50.4	3.6	21.1
Brightness (+30%)	50.9	71.6	55.8	74.6
Contrast (+30%)	80.7	97.7	86.3	98.5

online deployment at scale. At $D = 10$, the search space grows to $2^{10} = 1024$ candidates, completing in 28.9ms (35ms total), still well within real-time requirements.

At $D = 16$ (our default for maximum robustness), the search space is $2^{16} = 65{,}536$ candidates. Our unoptimized Python implementation takes 1.7s on a single CPU core. With 8-way parallelism, this reduces to 432ms (438ms total). In a production cloud environment, the CRC32 checking alone achieves ∼500k checks/sec on a 224-bit input, meaning the 65k candidates at $D = 16$ would complete in ∼120ms with optimized C/Rust implementations. The choice of D thus provides a latency-robustness trade-off: $D = 6$ already achieves 95.8% accuracy under blur ($\sigma = 2$) and 99.4% on clean images; $D = 16$ improves crop (80%) from 48.3% to 73.2% at the cost of higher latency, making it suitable as a forensic-grade deep search mode.

C.8 Comparison Against a Conventional Channel Code

The robustness gains above are measured against a hard-decision image decoder. A natural question is how much of that gain a *conventional* error-correcting code would deliver given the same redundancy, and therefore what the cross-modal channel adds beyond standard coding. We construct four decoders at matched redundancy on Qwen3-VL (n=300), all reading the same 256-bit payload (224 data + 32 redundancy) from the same images, and differing only in how the 32 redundancy bits are spent: (A) hard decision, accepted only if bit-exact; (B) an *idealized* bounded-distance code that succeeds whenever the error weight is at most t, which upper-bounds any real BCH or LDPC code at this rate; (C) our Soft-OSD with CRC but no Text Oracle; and (D) full CoSignature.

Table 12: **Instantiating CoSignature with a Gumbel text watermarker** (Qwen3-VL). Only the text watermarker changes; the image watermarker and joint decoder are identical. KGW column reproduced from Table 2 for reference.

Metric	**KGW** (n=1000)	**Gumbel** (n=300)
Text TPR (system key)	99.8	93.3 $\pm$2.9
Image Exact Match	89.6	88.0 $\pm$3.7
Binding TPR	**99.0**	**95.3** $\pm$2.4
Swap: WM–WM	99.0	93.9
Swap: NoWM(img) + WM(txt)	99.8	93.3 $\pm$2.9
Swap: WM(img) + NoWM(txt)	99.0	99.0 $\pm$1.3
Avg. swap detection	**99.3**	**95.4**

Table 13: **Per-Sample Verification Latency.** Extract = image watermark extraction (GPU); OSD = candidate enumeration + CRC check (CPU).

D	**Extract**	**OSD (1 core)**	**OSD (8 cores)**	**Total (8 cores)**
6	6.2ms	1.8ms	—	8.0ms
10	6.2ms	28.9ms	—	35.0ms
16	6.2ms	1,705ms	432ms	438ms

Matched redundancy alone does not make these decoders comparable, because t is an operating point rather than a free parameter. A t-error-correcting code accepts every word within Hamming distance t of a codeword, and those balls are disjoint, so its false-accept rate on non-watermarked content is exactly $|\mathrm{ball}(t)|/2^{32}$: 7.7×10^{-6} at t=2 but 4.1×10^{-2} at t=4, the largest t the Hamming bound permits. Our decoder filters the 58,651 candidates enumerated at D=16 through a 32-bit CRC, giving 1.4×10^{-5}. The comparable operating point is therefore t=2; a t=4 decoder is 3,000$\times$ more permissive and would assign a confident Content ID to one in every 24 unwatermarked images, which no provenance verifier can tolerate.

Table 14: **Comparison against an idealized channel code at matched redundancy.** Content ID recovery (%) on Qwen3-VL. Each decoder is labelled with its false-accept rate on non-watermarked content; only the t=2 column is comparable to CoSignature. Δ is CoSignature's margin over the FAR-matched code. All four arms decode the same n=300 subsample, so absolute values differ slightly from Table 3 (n=1000); the comparison between arms is unaffected.

Attack	**Img only** (bit-exact)	**ECC** t=2 7.7×10^{-6}	**ECC** t=4 4.1×10^{-2}	**CoSignature** 1.4×10^{-5}	Δ vs t=2
Clean	88.0	100.0	100.0	**100.0**	0.0
Blur (σ = 2)	89.7	99.7	100.0	**99.7**	0.0
JPEG (Q=75)	75.3	97.0	99.0	**98.3**	+1.3
Crop (90%)	61.0	92.0	98.0	**97.0**	+5.0
JPEG (Q=50)	36.3	77.3	88.0	**84.0**	+6.7
Crop (80%)	14.0	49.0	70.3	**63.3**	**+14.3**
Noise (σ = 10)	12.3	38.0	54.7	**50.0**	**+12.0**

Table 14 shows that at the matched operating point the joint decoder is never worse than the idealized code, winning on five of seven distortions and tying on the two mildest, and that its margin grows as the image channel degrades. The reason is structural: a bounded-distance decoder repairs any error pattern of weight $\leq t$ wherever it falls, a ball of fixed size, whereas the soft-decision search repairs any pattern confined to the D least reliable positions—and as distortion grows, errors concentrate in exactly those low-confidence positions. Where the channel is nearly clean the two coincide, because there is little to repair.

We also report arm (C) for transparency. At C=32 arms (C) and (D) are identical to within measurement precision on every distortion. This is a property of the operating point rather than a null result: 58,651 candidates against a 32-bit CRC leaves 1.4×10^{-5} expected false survivors, so $|\mathcal{K}_{cand}|$ is a singleton almost surely and the Text Oracle confirms rather than selects. At this checksum width the

recovery gain over a hard decision is produced by the CRC-filtered soft-decision search. The next section isolates the regime in which the oracle's selection role becomes load-bearing.

C.9 The Text Channel Substitutes for Checksum Redundancy

Section C.8 shows that a wide checksum leaves the oracle nothing to disambiguate. This is the regime in which no redundancy is scarce, and it is not the regime that matters in practice: many deployed image watermarkers carry far fewer than 256 bits, and reserving 32 of them for a checksum is often unaffordable. We therefore hold the search depth fixed and vary only the check width C, so that the number of enumerated candidates is unchanged and the expected number of *false* survivors of the check grows as $58{,}651/2^C$.

Table 15: **The text channel substitutes for checksum redundancy.** Content ID recovery (%) on Qwen3-VL (n=200) as the check width C shrinks at fixed search depth. The OSD recall is the rate at which the true payload is present among the candidates at all, and is identical across check widths, so these rows isolate disambiguation. Shortening the check returns $32 - C$ bits to Content ID capacity.

Configuration	$\mathbb{E}$[false survivors]	**Clean**	**JPEG (Q=50)**	**Crop (80%)**
OSD recall (ceiling)	—	100.0	83.0	64.0
C=32 (224-bit ID), CRC only	1.4×10^{-5}	100.0	83.0	64.0
+ Text Oracle		**100.0**	**83.0**	**64.0**
C=16 (240-bit ID), CRC only	0.89	38.0	35.0	27.5
+ Text Oracle		**100.0**	**83.0**	**64.0**
C=8 (248-bit ID), CRC only	229	0.0	0.0	0.5
+ Text Oracle		**98.0**	**81.0**	**62.5**

Table 15 answers the question posed in Section 1 quantitatively. At C=16 the check leaves more than one survivor in 68–90% of decodes and a check-only decoder falls to 27.5–38.0%, while the oracle recovers the full ceiling. At C=8 the check retains almost no discriminating power, leaving ≈232 survivors on average (closely matching the predicted 229) and a check-only recovery rate of 0.0–0.5%; the oracle nonetheless selects the true payload from those 232 candidates at 98.0, 81.0 and 62.5%, which is 97.6–98.0% of the achievable ceiling in every case. The text channel thus performs the disambiguation a 32-bit CRC performs, using 8 bits, and the 24 bits saved return directly to Content ID capacity. This is the sense in which one modality substitutes for redundancy the other cannot afford, and it is a capability unavailable to any unimodal decoder: the oracle ranks candidates on a statistic carried entirely by the text.

C.10 Security Analysis Details

An attacker attempting to forge a binding must know both k_{cont} (embedded in image pixels, requiring the private decoder) and k_{sys} (secret system key with dynamic per-token state). We test these barriers empirically with an adaptive black-box attacker who observes B watermarked articles sharing k_{sys}, holds a target watermarked image but not the private image decoder, and receives only a binary verdict. Granting the attacker the optimistic rule that every observed bigram is green, green-list inference on k_{sys} saturates at ≈38% precision (chance 25%) even at B=800, because the green list is re-seeded by the preceding token at each step, so coverage of specific contexts grows far more slowly than the query count.

Text constructed to maximize inferred-green hits passes *source* detection at 99.5–100%. We state this plainly: any watermark with a *reusable* key is spoofable given query access, and k_{sys} is reusable by design. The same text passes *binding* verification for only 0.0–2.5% of attempts across $B \in \{50, 200, 800\}$, statistically consistent with the 1% target FPR, because k_{cont} is per-article, never reused, and recoverable only through the private image decoder. This is precisely the separation the dual-key design is intended to provide. The result also gives the empirical justification for conditioning the null on $\hat{p}_{sys}$ (Eq. 5): under an unconditional null $\mu_w = N\gamma(1+\gamma)$ the identical attack succeeds 12–19% of the time, because $\mathcal{G}_{sys}$-enriched text inflates the double-green term for *every* candidate key and the score crosses threshold without the attacker knowing k_{cont}.

We view the security of our watermarking system in CoSignature as a subset of information hiding Anderson (2020). The watermark system should contain enough information for a verifier to ensure a high probability of success from a legitimate system while filtering out untrusted content from an attacker. In CoSignature, we have a linked high throughput channel in the image watermark with a low throughput channel in the text watermark. Building on the implied threat model from Jovanović et al. (2025), we assume a powerful attacker with control of the image-text pair after generation and will attempt transformations and generations to force a victim to accept untrusted data including misinformation. Under those requirements, the goals of the attacker against CoSignature are to forge, unbind, rebind, remove, detect, and recover the bound watermark.

With this threat model, the primary security goal of CoSignature is to discriminate between trusted generated image-text pairs and untrusted pairs. We assume that an attacker is powerful enough to recover k_{cont} using steps 1 and 2 from our joint source-channel decoder scheme, but that is as far as they can go. The k_{sys} remains secret, so the second Green List $G_{sys}^{(t)}$ stays unknown to the attacker. They can brute force a possible list, narrowing by likely candidate sets, but as long as the size of $G_{sys}^{(t)}$ generated from k_{sys} is small with a correspondingly small N_{min}, the likelihood that the attacker will correctly pick a token from the hidden $G_{sys}^{(t)}$ every time decreases exponentially with the length of text. Specifically against the threat model, our system is robust along the following attacks.

1. Forgery: In order to be able to forge a CoSignature, the attacker will need to know k_{sys} in addition to recovering successfully recovering k_{cont} from the image pixel data. Without k_{sys}, $G_{sys}^{(t)}$ stays unknown to the attacker and prevents them from generating a forged text watermark on text they control.
2. Rebind and unbind: Rebinding is a direct application of forgery which CoSignature is resistant to as long as k_{sys} remains secret. Unbinding is possible but only results in generating untrusted data. The user can correctly identify the modified data as untrusted and filter that out.
3. Removal. This is a generalization of the "Unbind" case and will also result in untrusted data that can be filtered out by the user.
4. Detection. The attacker can only recover k_{cont} from the image pixel data if it has access to the private image decoder with the verifier's private key, also the attacker cannot detect the binding with the text unless they also have k_{sys}.
5. Recovery: This attack could assist the forgery attack, but CoSignature keeps k_{sys} secret and prevents the full recovery of the watermark. Even if the image-text binding was detected, it only reveals a boolean from the text and does not reveal the static system key needed for forgery.

D Broader Impacts

The paper proposes a cross-modality watermarking scheme that establishes verifiable bindings between generated image and text modalities. However, we acknowledge that widespread adoption of provenance tracking carries inherent risks, including potential false detection. Ultimately, we believe that the safeguards introduced by verifiable cross-modality binding provide a necessary layer of accountability for the next generation of generative media, ensuring that the provenance of the *pairing* is as verifiable as the assets themselves.