BAYESIAN INFERENCE FOR ROBUST VIDEO WATERMARKING

Wonhyuk Ahn, Jihyeon Kang, Seung-Hun Nam
NAVER WEBTOON AI
{whahnize,mangorism,shnam1520}@gmail.com

ABSTRACT

We propose a simple yet effective *Bayesian extractor* for multi-frame video watermarking that can be plugged into any existing *image-based* watermarking method, such as HiDDeN, CIN, MBRS, TrustMark, WAM, or VideoSeal. In particular, we focus on challenging real-world conditions where videos undergo *repeated* or *strong* compression (e.g., H.264, H.265) or frame-rate changes that typically degrade watermark signals severely. When all frames carry the same hidden bits, our Bayesian extractor treats each frame's output as an independent observation and aggregates the log-likelihood ratios across frames, in contrast to naive averaging. Despite only modifying the extraction phase, this approach consistently boosts bit accuracy under moderate-to-aggressive compression, frame-rate conversions, and other distortions—while preserving the same watermark imperceptibility and embedding efficiency as the baseline. Experiments on diverse transformations and watermarking models show that these benefits are particularly pronounced when frames encounter uneven or heavy distortions, making our Bayesian extraction a lightweight but potent upgrade for robust video watermarking.

1 INTRODUCTION

Ensuring authenticity and traceability of digital videos is increasingly important, especially as online platforms frequently *re-encode* or *downsample* user uploads. Deep learning-based *image* watermarking has made significant progress in embedding robust and imperceptible signals into static images (Zhu, 2018; Jia et al., 2021; Bui et al., 2023; Sander et al., 2024). However, applying these methods directly to *video* highlights additional challenges:

1. **Repeated Compression:** Popular streaming services (YouTube, Vimeo) and social media apps (Instagram, TikTok) commonly re-encode uploaded videos multiple times at different bitrates or resolutions, which can drastically weaken or remove watermark signals.
2. **Frame-Rate Changes:** Videos recorded at one frame rate (e.g., 24 fps) may be transcoded to 12 fps, 30 fps, or 60 fps, creating uneven or unpredictable distortions frame by frame (Wang et al., 2022).
3. **Heavy Computation & Real-Time Constraints:** Watermarking each frame individually can be computationally heavy for long videos, and real-time scenarios often have strict latency requirements.

A common baseline approach is to *embed the same bits* in each frame and then aggregate the extracted signals across frames (Zhu, 2018; Fernandez et al., 2024). However, this aggregation is often done via naive averaging of probabilities, which fails to adequately discount heavily corrupted frames. Our primary contribution is a *Bayesian extraction* strategy that:

- **Summation of Log-Likelihood Ratios:** Instead of naive averaging, each frame's output (logits) is converted into a log-likelihood ratio (LLR), then summed across frames. This approach inherently downweights uncertain frames, providing more robust bit recovery under heavy compression or partial corruption.
- **No Changes to Embedding:** Our method requires no modifications to existing 2D watermarking pipelines—such as HiDDeN (Zhu, 2018), CIN (Ma et al., 2022), MBRS (Jia et al.,

2021), TrustMark (Bui et al., 2023), WAM (Sander et al., 2024), or VideoSeal (Fernandez et al., 2024)—thus serving as a simple plug-and-play upgrade in the extraction phase.

- **Better Handling of Frame-Level Variance:** By weighting frames according to their confidence, the Bayesian aggregator becomes more reliable precisely when distortions vary across frames, such as with repeated compression or frame-rate transformations.

Through experiments on five-second clips from the SA-V dataset (Ravi et al., 2024), we find that Bayesian extraction—via log-likelihood summation—consistently outperforms naive averaging. Specifically, our method recovers more bits under repeated or strong compression (e.g., high CRF values, multiple re-encodes) and shows marked gains when the output frame rate diverges from the original (e.g., 24 fps to 12 fps). Under uniform, mild distortions, improvements are modest but still positive. These results indicate that applying Bayesian aggregation during extraction alone substantially enhances watermark robustness in real-world video pipelines, where re-encoding and frame-rate changes are common.

2 Related Work

Image Watermarking. Deep watermarking for images has been widely studied (Zhu, 2018; Jia et al., 2021; Bui et al., 2023), typically treating each image independently. These methods demonstrate strong resilience to typical image-level edits (JPEG compression, mild rotations, color changes), but few handle advanced geometric or partial inpainting scenarios (Sander et al., 2024). In principle, any of these 2D approaches can be extended to a video by embedding the same bits in each frame.

Video Watermarking Extensions. Adapting image-based watermarking networks to video has been approached via 3D CNNs (Luo et al., 2023) or by reshaping frames into pseudo-batches (Ye et al., 2023). Nonetheless, extending image-based watermarking to video has become attractive for two reasons: (1) the lightweight nature of these models allows high-resolution videos to be watermarked, and (2) well-defined image watermarking models can be leveraged directly for video applications (Fernandez et al., 2024). Our *Bayesian aggregator* summation is a purely extract-time improvement that is orthogonal to the embedding process.

Temporal Watermark Propagation. Watermarking *each* frame can be computationally expensive for long videos. Xian et al. (2024) suggest a shortcut of watermarking every k frames, leaving other frames un-watermarked, but this can reduce extraction accuracy. Fernandez et al. (2024) propose *temporal watermark propagation*: they embed once every k frames and *copy* the resulting watermark distortion to the $k-1$ subsequent frames. This is a process that preserves overall consistency and helps mitigate per-frame cost while keeping all frames watermarked. In this paper, we adopt the same temporal propagation strategy at inference time to produce a watermarked video, reusing the original model weights from existing 2D networks.

3 Method: Bayesian Extraction with Temporal Propagation

Preliminary Notation. Throughout this section, we designate the networks for watermark *embedding* and *extraction* as E and D, respectively. Each originally operates on a single image $x \in \mathbb{R}^{3\times256\times256}$, embedding or extracting N bits.

3.1 Frame-Wise Embedding

We reuse the **model weights** from open-source watermarking networks (e.g., VideoSeal (Fernandez et al., 2024), CIN (Ma et al., 2022), HiDDeN (Zhu, 2018), MBRS (Jia et al., 2021), TrustMark (Bui et al., 2023), WAM (Sander et al., 2024)).

$$w_i = \begin{cases} E\big(x_i, m\big), & \text{if } i \mod k = 0, \\ w_{i-1}, & \text{otherwise,} \end{cases} \tag{1}$$

where w_i is the watermark distortion for the ith frame, $x_i \in \mathbb{R}^{3\times256\times256}$ is the ith (resized) frame, and $m \in \{0,1\}^N$ is the bitstring.

We then add w_i to x_i (optionally scaling by a factor α). Unless otherwise stated, we set $k = 4$, balancing cost and consistency. Hence, *every* frame is watermarked, though only *one out of every* k frames is passed through the embedder E.

3.2 Bayesian Multi-Frame Extraction

Naive Averaging (Baseline). Averaging simply combines per-frame probabilities $p_{i,j}$ (for bit index j) across all T frames:

$$\hat{m}_j^{(\text{avg})} = \mathbf{1}\left[\frac{1}{T}\sum_{i=1}^{T} p_{i,j} > 0.5\right], \tag{2}$$

where $p_{i,j} = \sigma(\tilde{m}_{i,j})$ is the predicted probability from the extractor's logit $\tilde{m}_{i,j}$. While this often outperforms single-frame extraction, heavily corrupted frames (near-random logits) can skew the average.

Bayesian Extraction (Ours). During extraction, we treat each frame *independently* through the original 2D extractor network D, yielding a *logit vector* $\tilde{m}_i \in \mathbb{R}^N$ per frame. Unlike naive averaging, which simply sums probabilities, the **Bayesian aggregator** reinterprets each frame's logit as a log-likelihood ratio (LLR), then *sums* across frames:

$$\text{LLR}_{i,j} \;=\; \log\Big(\frac{p_{i,j}}{1-p_{i,j}}\Big), \quad \text{LLR}_j^{(\text{sum})} \;=\; \sum_{i=1}^{T} \text{LLR}_{i,j}. \tag{3}$$

Finally, we threshold each $\text{LLR}_j^{(\text{sum})}$ at zero to obtain the predicted bit:

$$\hat{m}_j^{(\text{Bayes})} = \mathbf{1}\big[\text{LLR}_j^{(\text{sum})} > 0\big]. \tag{4}$$

Because uncertain frames (where $p_{i,j} \approx 0.5$) contribute near-zero LLR, the aggregator naturally discounts heavily corrupted frames.

4 Experimental Setup

Dataset for Inference. We follow Fernandez et al. (2024) and select the first 5 seconds clip from the SA-V dataset (Ravi et al., 2024), covering diverse real-world conditions (240p–4K, various scenes). *We do not perform any training ourselves*, since we rely on the **pretrained weights** of each watermarking model. We simply run temporal propagation ($k = 4$) to embed the same bits in these 5-second clips, then apply either the *Bayesian* or *naive* extraction method for all models and transformations.

Baseline Models. Following VideoSeal (Fernandez et al., 2024), we use state-of-the-art image watermarking models as baselines for video watermarking. In particular, we consider HiDDeN (Zhu, 2018) (48 bits), MBRS (Jia et al., 2021) (256 bits), CIN (Ma et al., 2022) (30 bits), TrustMark (Bui et al., 2023) (100 bits), and WAM (Sander et al., 2024) (32 bits), using the official pretrained weights from VideoSeal for all except HiDDeN. VideoSeal itself operates with 96 bits and a watermark strength of $\alpha_w = 2.0$. All methods are applied at 256×256 (with CIN at 128×128, extended as needed) and use temporal watermark propagation with $k = 4$, ensuring a fair comparison with our proposed Bayesian extraction.

Transformations. We follow the transformations described in VideoSeal (Fernandez et al., 2024) (see Table 1). Additionally, we conduct experiments on frame-rate conversion and re-compression to better reflect real-world scenarios, as detailed in the Experiments section.

5 Results and Analysis

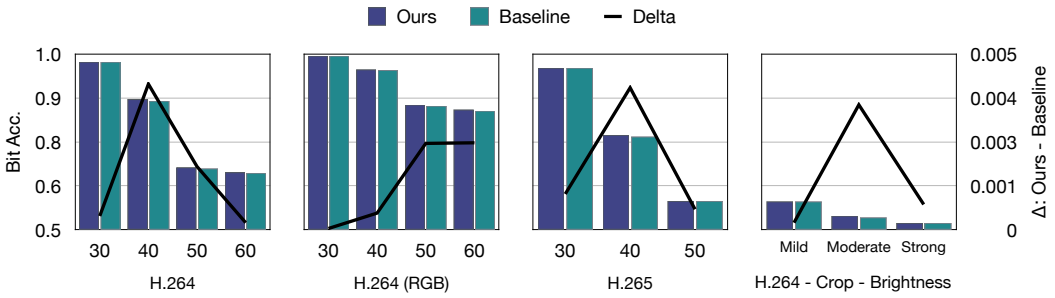

Figure 1: **Bit accuracy comparison** of the naive average, Bayesian aggregator, and Δ (*Ours*–Baseline) for H.264, H.265, and combined transformations across all models: CIN, HiDDeN, MBRS, TrustMark, WAM, and VideoSeal.

Table 1: Transformation types and parameters (Fernandez et al., 2024)

	Transformation	**Parameters**
Identity	-	-
H.264	Video Compression	CRF=30, 50, 60
H.264 (RGB)	Video Compression	CRF=30, 40, 50, 60
H.265	Video Compression	CRF=30, 40, 50
JPEG	Valuemetric	Quality=40
Horizontal Flip	Geometric	-
Rotate	Geometric	10, 90
Resize	Geometric	0.55, 0.71
Crop	Geometric	0.55
Perspective	Geometric	0.5
Brightness	Valuemetric	1.5
Contrast	Valuemetric	0.5, 1.5
Saturation	Valuemetric	0.5, 1.5
Hue	Valuemetric	0.25
Gaussian Blur	Valuemetric	Kernel=9
Median Filter	Valuemetric	Kernel=9
H.264 + Crop + Brightness	Combined	(CRF=30, 0.71, 0.5)
H.264 + Crop + Brightness	Combined	(CRF=40, 0.71, 0.5)
H.264 + Crop + Brightness	Combined	(CRF=50, 0.71, 0.5)

We focus our evaluation on **video compression** scenarios, where repeated encoding (e.g., H.264, H.265) often degrades watermarks severely. Since we do not alter the embedding process and instead reuse pretrained weights (§3.1), the *watermark imperceptibility* (, PSNR, SSIM, LPIPS, VMAF) and *embedding complexity* follow the original baselines Fernandez et al., 2024. Furthermore, the overhead from our Bayesian extractor at inference is minimal (nearly identical to naive extraction).

Compression Results Across All Models. As illustrated in Fig. 1, our Bayesian extraction method consistently outperforms naive averaging across various compression settings—including H.264, H.264rgb, and H.265—when applied to all evaluated watermarking models (CIN, HiDDeN, MBRS, TrustMark, WAM, and VideoSeal). Although the absolute improvements in bit accuracy are modest, they are consistent and most pronounced at moderate-to-strong compression levels. These results confirm that our Bayesian aggregator provides a systematic advantage in reliably extracting watermark bits from videos subjected to diverse compression transformations across all state-of-the-art image watermarking models.

Model-Agnostic Gains. Figure 2(a) shows that Bayesian extraction yields consistent improvements across diverse watermarking models. While some models (e.g., CIN, HiDDeN, MBRS) exhibit only marginal gains (with Δ on the order of 0.0001–0.0005), others such as TrustMark and WAM see larger improvements (e.g., $\Delta \approx 0.0027$ and 0.0057, respectively). These model-agnostic gains demonstrate that the Bayesian extraction method is a broadly applicable, drop-in enhancement for various watermarking systems.

Frame Rate Conversion. In practice, videos are often re-encoded at different frame rates (e.g., a 24 fps video uploaded to YouTube may be converted to 60 fps). As Fig. 2(b) shows, when the embedding rate remains at 24 fps, Bayesian extraction offers little gain. However, under frame-rate conversion (e.g., 24 fps to 12 fps or 60 fps), its advantage grows. For example, at 12 fps the mean bit accuracy increases from 0.7656 to 0.7792, compared to only a +0.0010 gain at 24 fps. This demonstrates that Bayesian aggregation is especially beneficial when frame-rate changes disrupt temporal consistency.

Temporal Propagation Step Size Ablation. As shown in Fig. 2(c), a smaller step size (e.g., $k = 2$)—with more frequent embedding and fewer propagated frames—yields a larger gain (around $\Delta \approx 0.0038$). In contrast, increasing k (to 4 or 8) introduces more propagation noise, reducing the gain

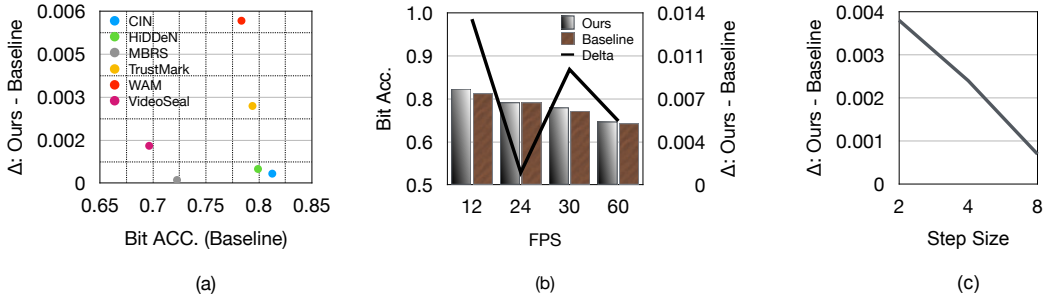

Figure 2: (a) Bit accuracy and Δ for each model under video compression transformations. (b) Bit accuracy and Δ for WAM under frame-rate conversion using H.264 (CRF=40) with embedding at 24 fps. (c) Relationship between step size k and Δ for VideoSeal under H.265 (CRF=40) compression.

to $\Delta \approx 0.0024$ and $\Delta \approx 0.0007$, respectively. Thus, Bayesian aggregation is most effective when fewer frames are independently watermarked.

Re-compression Results. In real-world, videos undergo multiple, unintentional compression steps—for example, when saving a video, uploading it to a platform like YouTube, and then downloading it. Although these operations are not malicious, each re-encoding stage can severely degrade the embedded watermark. Table 2 shows results for a three-stage re-compression pipeline. With each added stage, naive averaging drops further, while Bayesian extraction gains 0.0031 and 0.0115 bit accuracy at stages 2 and 3, respectively. We hope this showcase motivates future research into robust watermarking methods that can better handle such multi-stage, non-adversarial re-compression pipelines.

Step	Compression	Baseline	Ours	Δ
1	H.264 (CRF=23, FPS=24)	0.9990	0.9990	0.0000
2	+ H.265 (CRF=24, FPS=60)	0.9688	0.9719	0.0031
3	+ H.264 (CRF=30, FPS=30)	0.9052	0.9167	0.0115

Table 2: Cumulative re-compression results for WAM. Step 1 is the initial compression; subsequent steps (denoted by '+') apply additional re-compression.

Other Results. Table 3 summarizes the baseline bit accuracy using naive averaging and the corresponding improvements achieved by our Bayesian extractor for non-compression transformations (Identity, Valuetric, Geometric). The gains in these scenarios are generally modest, reflecting that such transformations are often mild or uniformly applied across frames. This indicates that our primary advantage is realized under uneven or strong distortions (e.g., compression). Overall, these results confirm that Bayesian extraction is especially beneficial when frames experience *uneven* or *strong* distortions, such as repeated compression, while maintaining at least comparable performance in milder conditions.

Model	Identity		Valuetric		Geometric	
	Bit Acc.	Δ	Bit Acc.	Δ	Bit Acc.	Δ
HiDDeN	0.997	+1e-4	0.983	+0	0.776	+0
MBRS	1.000	+0	0.994	+0	0.627	+2e-4
CIN	1.000	+0	1.000	+0	0.626	+0
TrustMark	1.000	+0	0.998	+1e-4	0.700	+0
WAM	1.000	+0	0.998	+1e-4	0.855	+1e-4
VideoSeal	0.988	+0	0.984	+0	0.870	+3e-4

Table 3: Baseline bit accuracy (Bit Acc.) and improvement (Δ) from Bayesian extraction for non-compression transformations.

6 Conclusion

We have presented a *Bayesian extraction* method for multi-frame watermarking that is entirely plug-and-play: it requires no modifications to existing 2D embedding models, yet it can yield consistent gains in robustness, particularly when different frames are subject to uneven or repeated distortions. Under mild edits, improvements may be small; however, under more pronounced compression or partial corruption, the difference can be significant. Our experiments on diverse transformations confirm the generality of this approach as a simple yet valuable upgrade for video watermarking. We hope these findings encourage further research into multi-frame inference strategies that enhance the reliability of deep watermarking in real-world, compressed, and frame-varying video pipelines.

REFERENCES

Tu Bui, Shruti Agarwal, and John Collomosse. Trustmark: Universal watermarking for arbitrary resolution images. *arXiv preprint arXiv:2311.18297*, 2023.

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

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

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

Rui Ma, Mengxi Guo, Yi Hou, Fan Yang, Yuan Li, Huizhu Jia, and Xiaodong Xie. Towards blind watermarking: Combining invertible and non-invertible mechanisms. In *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 1532–1542, 2022.

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

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

Yifei Wang, Qichao Ying, Yunlong Sun, Zhenxing Qian, and Xinpeng Zhang. A dtcwt-svd based video watermarking resistant to frame rate conversion. In *2022 International Conference on Culture-Oriented Science and Technology (CoST)*, pp. 36–40. IEEE, 2022.

Xun Xian, Ganghua Wang, Xuan Bi, Jayanth Srinivasa, Ashish Kundu, Mingyi Hong, and Jie Ding. Raw: A robust and agile plug-and-play watermark framework for ai-generated images with provable guarantees. *arXiv preprint arXiv:2403.18774*, 2024.

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

J Zhu. Hidden: hiding data with deep networks. *arXiv preprint arXiv:1807.09937*, 2018.