DIFFUSIONSHIELD: A WATERMARKING APPROACH TO SAFEGUARDING VIDEO INTEGRITY AGAINST STABLE DIFFUSION

Zuzanna Gawrysiak[*], **Mateusz Gabor**[†], **Tomasz Hawro**
Vestigit
Wrocław, Poland
{zuzanna.gawrysiak, mateusz.gabor, tomasz.hawro}@vestigit.com

ABSTRACT

To address the critical safety and reliability challenges posed by the rapid advancement of Deep Generative Models (DGMs), we propose a principled video watermarking strategy designed to withstand diffusion-based tampering. While traditional forensic methods fail against the semantic reconstruction capabilities of modern DGMs, our approach bridges this gap by integrating a computationally efficient architecture with a differentiable attack simulation layer. We demonstrate the efficacy of this attention-guided framework through extensive experiments, achieving state-of-the-art fidelity (38.9 dB PSNR) and precise localization of Stable Diffusion inpainting (F1 0.85). This work provides a robust, scalable mechanism for ensuring media integrity, directly supporting the responsible deployment of generative AI.

1 INTRODUCTION

Video technology is currently one of the most heavily used forms of media in almost every aspect of human life, from social media feeds and news broadcasts to courtroom evidence and security surveillance (Verdoliva, 2020). However, this widespread presence has turned video content into a leading target for malicious manipulations, such as video clipping, tampering, or deep fakes. In recent years, the landscape of video forgery has shifted dramatically from manual editing tools to sophisticated deep learning algorithms.

While deep learning methods now drive the most advanced image and video forgery techniques (Yu et al., 2019; Chang et al., 2019; Xia et al., 2025; Wu et al., 2023; Jiang et al., 2025), they also provide the foundation for powerful defense mechanisms (Cozzolino et al., 2018; Wang et al., 2020; Guillaro et al., 2023; Rossler et al., 2019; Xu et al., 2023; Choi et al., 2024). To further protect content from modern generative threats, recent proactive defenses have introduced techniques to prevent style mimicry (Shan et al., 2023), mitigate the risks of AI-powered editing (Salman et al., 2023), and block unauthorized 3D reconstructions (Song et al., 2024). Specifically, deep learning-based watermarking has emerged as a key solution for content authentication. Among many deep learning approaches, we can distinguish those based on watermarking, such as RWVMD (Zhou et al., 2022) and VideoFACT (Nguyen et al., 2024), which offer a proactive defense mechanism against manipulation by securing the content at its source.

2 PROPOSED METHOD

The proposed DiffusionShield framework is designed to survive traditional video compression (e.g., H264, H265 codecs) and to detect localization of modern generative tampering (e.g., Stable Diffusion inpainting). As illustrated in Figure 1, the proposed pipeline operates in three stages (see more detailed figure in Appendix A). In the first stage, the generated watermark is embedded into

[*]Poznań University of Technology
[†]Wrocław University of Science and Technology

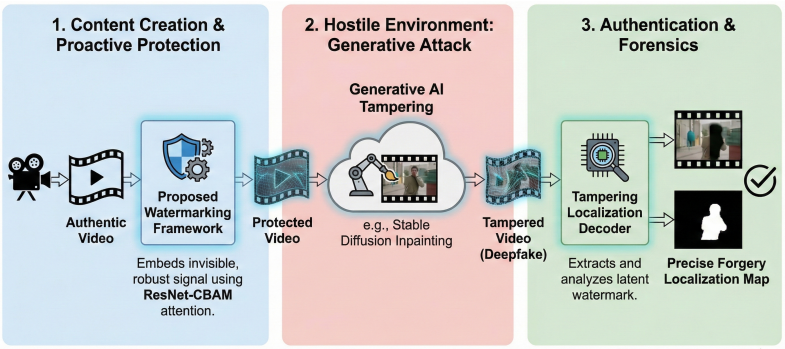

Figure 1: High-level workflow of the proposed watermarking framework.

the source video. In the second stage, the simulated generative attacks are performed. Finally, in the third stage, an accurate localization map is derived from the watermark to verify the authenticity of the content.

2.1 Architecture Overview

The DiffusionShield model consists of two networks: embedder (emb_θ) and detector (det_ϕ), where θ and ϕ denotes the weights of the embedder and detector, respectively.

Both the embedder emb_θ and the detector det_ϕ are based on a U-Net–derived encoder–decoder architecture with the residual connections and the Convolutional Block Attention Modules (CBAM). See Appendix B for a detailed description of the architecture.

The embedder maps an input frame $\mathcal{X} \in \mathbb{R}^{C\times H\times W}$ to a watermarked frame $\mathcal{Z} \in \mathbb{R}^{C\times H\times W}$ via

$$\mathcal{Z} = \mathrm{emb}_\theta(\mathcal{X}) = \mathcal{X} + \mathrm{Up}_{H\times W}\Big(\frac{\pi}{2}\arctan\Big(\mathcal{U}_\theta\big(\mathrm{Down}_{H'\times W'}(\mathcal{X})\big)\Big)\Big), \tag{1}$$

where $\mathrm{Down}_{H'\times W'}(\cdot)$ denotes bilinear downsampling to $H'\times W'$ and $\mathrm{Up}_{H\times W}(\cdot)$ denotes bilinear upsampling to $H\times W$. The backbone $\mathcal{U}_\theta$ is a U-Net–style encoder–decoder, and $\frac{\pi}{2}\arctan(\cdot)$ bounds the predicted residual watermark.

After embedding, the watermarked frame $\mathcal{Z}$ may be subjected to a arbitrary tampering operation $\mathcal{T}(\cdot)$, yielding the tampered watermarked frame $\hat{\mathcal{Z}} = \mathcal{T}(\mathcal{Z})$.

The detector det_ϕ takes as input a tampered watermarked frame $\hat{\mathcal{Z}}$ and outputs a predicted tampering mask $M \in \{0,1\}^{H\times W}$:

$$M = \mathrm{det}_\phi(\hat{\mathcal{Z}}) = \sigma\Big(\mathrm{Up}_{H\times W}\Big(\mathcal{U}_\phi\big(\mathrm{Down}_{H'\times W'}(\hat{\mathcal{Z}})\big)\Big)\Big), \tag{2}$$

where $\mathrm{Down}_{H'\times W'}(\cdot)$ denotes bilinear downsampling to $H'\times W'$ and $\mathrm{Up}_{H\times W}(\cdot)$ denotes bilinear upsampling back to $H\times W$, $\mathcal{U}_\phi$ is a U-Net–style encoder–decoder backbone, and $\sigma(\cdot)$ is the sigmoid function.

During training, for each applied tampering transform $\mathcal{T}$ we define a corresponding ground-truth binary tampering mask $M^{\mathrm{gt}} \in \{0,1\}^{H\times W}$ and generate a tampered watermarked frame by splicing content from an irrelevant source. Concretely, given a watermarked frame $\mathcal{Z}$ and an irrelevant frame $\mathcal{Z}_{\mathrm{irr}}$ (content from a different video with distinct semantic features), we construct

$$\hat{\mathcal{Z}} = \mathcal{T}\big(\mathcal{Z};\, M^{\mathrm{gt}}, \mathcal{Z}_{\mathrm{irr}}\big) = \big(1 - M^{\mathrm{gt}}\big)\odot\mathcal{Z} \;+\; M^{\mathrm{gt}}\odot\mathcal{Z}_{\mathrm{irr}}, \tag{3}$$

where $\odot$ denotes element-wise multiplication. The network is then trained in a supervised way using the pair (M, M^{gt}).

The framework is trained end-to-end by minimizing a joint objective $\mathcal{L}_{\mathrm{total}} = \lambda\,\mathcal{L}_{\mathrm{fid}} + \mathcal{L}_{\mathrm{loc}}$, where the fidelity term $\mathcal{L}_{\mathrm{fid}}$ encourages imperceptible embedding, and the localization term $\mathcal{L}_{\mathrm{loc}}$ supervises tamper segmentation. The scalar $\lambda > 0$ controls the trade-off between embedding fidelity and localization accuracy. More details about the losses are provided in Appendix D.

2.2 DIFFERENTIABLE ATTACK SIMULATION

To ensure the watermarking framework is robust against both malicious tampering and benign transmission distortions, we are using a differentiable compression layer Sander et al. (2024); Fernandez et al. (2024) between the encoder and decoder to enforce watermark robustness against video compression. Given a batch of tampered watermarked frames $\mathcal{V} = \{\hat{\mathcal{Z}}^{(b)}\}_{b=1}^{B} \in \mathbb{R}^{B\times C\times H\times W}$, with $\hat{\mathcal{Z}}^{(b)} \in \mathbb{R}^{C\times H\times W}$ and B as batch size. To emulate real-world transmission artifacts, the entire batch is then passed through an H.264/H.265 codec. Since standard codecs are non-differentiable, we adopt a straight-through estimator Fernandez et al. (2024); Bengio et al. (2013) and define the compressed output as:

$$\hat{V} = V + \mathrm{sg}(\mathrm{H.26x}(V) - V), \tag{4}$$

where $\mathrm{sg}(\cdot)$ is the stop-gradient operator (identity in the forward pass and zero-gradient in backprop) and $\mathrm{H.26x}$ denotes either H.264 or H.265 compression. During training, we apply H.264/H.265 compression with a bitrate of $4{,}000$–$8{,}000$, the medium preset, and a frame rate of 25 fps.

3 EXPERIMENTS

The training was carried out on a dataset created from DAVIS (Perazzi et al., 2016) and YouTube-VOS (Xu et al., 2018) training split. Evaluation is performed on the unseen YouTube-VOS test set. Following standard protocols in video watermarking (Zhou et al., 2022), we construct our training set using videos from the DAVIS (Perazzi et al., 2016) and YouTube-VOS (Xu et al., 2018) train splits. The proposed method was trained using the AdamW optimizer for 100 epochs with a learning rate of 1×10^{-3} and a weight decay of 1×10^{-4} on a single NVIDIA H100 GPU. The framework was implemented by using *PyTorch* library.

The evaluation of the proposed method was tested against robustness against both manipulation and generative attacks on the unseen YouTube-VOS test set. Furthermore, we analyze the resilience of the generated watermark under various video compression settings. As competing approaches, we selected two representative methods: RWVMD (Zhou et al., 2022), a state-of-the-art end-to-end video watermarking framework utilizing a 3D-UNet architecture, and VideoFACT (Nguyen et al., 2024), a recent passive forgery detection method designed to detect manipulations without watermarks.

To evaluate fidelity, we used Peak Signal-to-Noise Ratio (PSNR) on the test set. For tampering localization, we compute pixel-wise Precision, Recall, and F1 between the predicted mask M and the ground-truth mask M^{gt}.

3.1 ROBUSTNESS AGAINST MANIPULATION AND GENERATIVE ATTACKS

The primary contribution of this work is resilience against diffusion-based generative tampering. We simulate tampering using two state-of-the-art models: Stable Diffusion Inpainting (SD) (Rombach et al., 2022) and RePaint (Lugmayr et al., 2022).

As shown in Table 1, passive methods like VideoFACT struggle significantly with generative attacks, achieving an F1 score of only 0.35 on Stable Diffusion. This indicates that the statistical artifacts introduced by modern diffusion models are subtle enough to evade passive detectors. Similarly, the baseline RWVMD fails to reliably localize diffusion-inpainted regions, as its features are often destroyed by the semantic reconstruction process of diffusion models. In contrast, our approach achieves an F1 score of 0.8 against Stable Diffusion and 0.78 against RePaint. This confirms that the CBAM mechanism successfully locks onto robust semantic features that survive the generative reconstruction process. For traditional splicing attacks (copy-paste), our method matches the near-perfect performance of RWVMD (0.99 F1), ensuring no trade-off in standard security capability. Qualitative results are presented in Figure 2. As observed in the first row, our method successfully identifies inpainted regions, distinguishing the generated 'hallucinations' from the original video content despite the high semantic consistency of the attack. The visualization of the watermark itself is shown in Appendix C.

Our method achieves an average PSNR of 38.9 dB, significantly outperforming the 3D-UNet-based RWVMD baseline (37.8 dB). This improvement suggests that our lightweight ResNet-CBAM en-

Table 1: Localization accuracies of the proposed method against SD, RePaint, and traditional Splicing compared with VideoFACT and RWVMD; † as reported in Zhou et al. (2022).

Method	Stable Diffusion Inpaint			RePaint			Splicing		
	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall
RWVMD†	-	-	-	-	-	-	0.99	0.99	0.99
VideoFACT	0.35	0.25	0.88	0.35	0.25	0.87	0.35	0.26	0.88
DiffusionShield	0.85	0.98	0.79	0.78	0.98	0.73	0.99	0.99	0.98

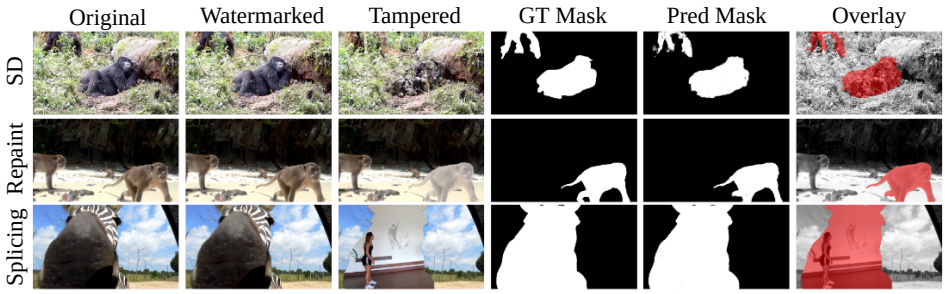

Figure 2: Qualitative visualization of tampering localization under generative and traditional attacks. The columns display the full pipeline: from the original and watermarked frames to the tampered input.

coder is more effective at embedding imperceptible signals into high-frequency spatial regions compared to computationally heavier volumetric approaches. Note that VideoFACT is a passive method and does not modify the source video.

3.2 Robustness to Video Compression

To verify that our method is suitable for real-world deployment, we evaluate robustness against standard lossy compression standards: H.264 (CRF 17) and H.265 (CRF 17).

Table 2 shows that our method is robust against video compression. While RWVMD experiences a performance drop under H.264 compression (F1 decreases to 0.87), our method sustains an F1 score of **0.96** for H.264 and **0.98** for H.265. This result addresses the concern that 2D architectures might lack temporal consistency; our training pipeline ensures that the proposed model remains robust even under inter-frame prediction coding used in modern video codecs.

Table 2: Localization accuracies of the proposed method against image and video compression with Splicing compared with VideoFACT and RWVMD; † as reported in Zhou et al. (2022).

Method	Clean			H.264 (CRF 17)			H.265 (CRF 17)		
	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recal
RWVMD†	0.99	0.99	0.99	0.87	0.95	0.83	-	-	-
VideoFACT	0.35	0.26	0.88	0.32	0.25	0.86	0.35	0.26	0.87
DiffusionShield	0.99	0.99	0.98	0.98	0.98	0.99	0.98	0.98	0.98

4 Conclusion

In this work, we presented a robust video watermarking framework DiffusionShield capable of authenticating media in the face of advanced generative attacks. By integrating a lightweight architecture with a differentiable attack simulation layer, our method achieves state-of-the-art imperceptibility (38.9 dB PSNR) while enabling precise tampering localization against both traditional splicing and modern diffusion-based inpainting compared to competitive approaches.

REFERENCES

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

Ya-Liang Chang, Zhe Yu Liu, Kuan-Ying Lee, and Winston Hsu. Free-form video inpainting with 3d gated convolution and temporal patchgan. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9066–9075, 2019.

Jongwook Choi, Taehoon Kim, Yonghyun Jeong, Seungryul Baek, and Jongwon Choi. Exploiting style latent flows for generalizing deepfake video detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1133–1143, 2024.

Davide Cozzolino, Justus Thies, Andreas Rössler, Christian Riess, Matthias Nießner, and Luisa Verdoliva. Forensictransfer: Weakly-supervised domain adaptation for forgery detection. *arXiv preprint arXiv:1812.02510*, 2018.

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

Fabrizio Guillaro, Davide Cozzolino, Avneesh Sud, Nicholas Dufour, and Luisa Verdoliva. Trufor: Leveraging all-round clues for trustworthy image forgery detection and localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20606–20615, 2023.

Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. *arXiv preprint arXiv:2503.07598*, 2025.

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11461–11471, 2022.

Tai D Nguyen, Shengbang Fang, and Matthew C Stamm. Videofact: detecting video forgeries using attention, scene context, and forensic traces. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 8563–8573, 2024.

Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 724–732, 2016.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1–11, 2019.

Hadi Salman, Alaa Khaddaj, Guillaume Leclerc, Andrew Ilyas, and Aleksander Madry. Raising the cost of malicious AI-powered image editing. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

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

Shawn Shan, Jenna Cryan, Emily Wenger, Haitao Zheng, Rana Hanocka, and Ben Y Zhao. Glaze: Protecting artists from style mimicry by {Text-to-Image} models. In *32nd USENIX Security Symposium (USENIX Security 23)*, pp. 2187–2204, 2023.

Qi Song, Ziyuan Luo, Ka Chun Cheung, Simon See, and Renjie Wan. Geometry cloak: Preventing tgs-based 3d reconstruction from copyrighted images. *Advances in Neural Information Processing Systems*, 37:119361–119385, 2024.

Luisa Verdoliva. Media forensics and deepfakes: an overview. *IEEE journal of selected topics in signal processing*, 14(5):910–932, 2020.

Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8695–8704, 2020.

Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7623–7633, 2023.

Bin Xia, Yuechen Zhang, Jingyao Li, Chengyao Wang, Yitong Wang, Xinglong Wu, Bei Yu, and Jiaya Jia. Dreamomni: Unified image generation and editing. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 28533–28543, 2025.

Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv preprint arXiv:1809.03327*, 2018.

Yuting Xu, Jian Liang, Gengyun Jia, Ziming Yang, Yanhao Zhang, and Ran He. Tall: Thumbnail layout for deepfake video detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 22658–22668, 2023.

Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4471–4480, 2019.

Yangming Zhou, Qichao Ying, Yifei Wang, Xiangyu Zhang, Zhenxing Qian, and Xinpeng Zhang. Robust watermarking for video forgery detection with improved imperceptibility and robustness. In *2022 IEEE 24th International Workshop on Multimedia Signal Processing (MMSP)*, pp. 1–6. IEEE, 2022.

A DETAILED PIPELINE

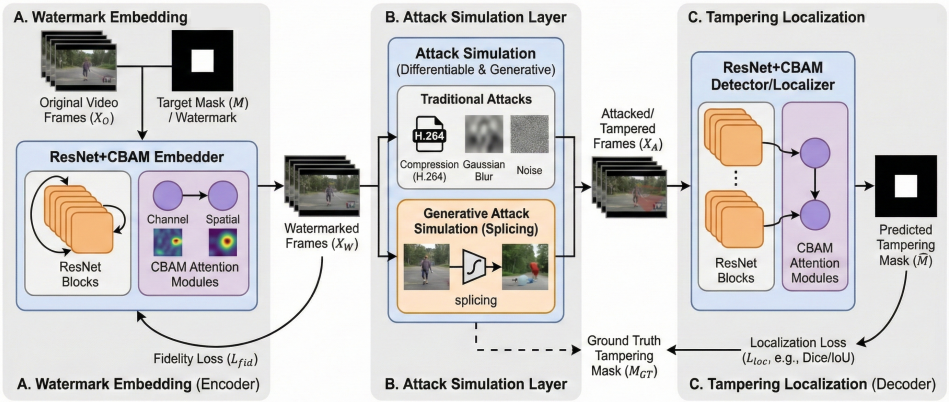

Figure 3: Schematic overview of the proposed DiffusionShield framework. The pipeline is trained end-to-end.

B ARCHITECTURE DETAILS

The proposed DiffusionShield framework utilizes a shared backbone architecture for both the embedder (emb_θ) and detector (det_ϕ). As detailed in Section 2, this backbone follows a ResUNet design equipped with Convolutional Block Attention Modules (CBAM). In this appendix, we detail the specific layer configurations, channel dimensions, and blocks used in our implementation.

B.1 BACKBONE SPECIFICATION

The core architecture, ResUNet Backbone, is an encoder-decoder network. The encoder contracts the feature map spatial dimensions while increasing channel depth, utilizing Residual Blocks (ResBlock) for feature extraction. The decoder recovers spatial resolution using bilinear upsampling, fusing features from the encoder via skip connections. Crucially, to enhance the localization of tampering artifacts, we integrate CBAM attention mechanisms specifically in the decoder path after feature fusion and convolution.

Table 3: Detailed architecture of the ResUNet Backbone used in DiffusionShield. C_{in} denotes the number of input channels (3 for RGB). The spatial resolution is halved at each Down stage and doubled at each Up stage.

Stage	Layers / Operations	Input Channels	Output Channels
	Encoder Path		
Stem	ResBlock	C_{in}	32
Down 1	MaxPool $(2 \times 2) \rightarrow$ ResBlock	32	64
Down 2	MaxPool $(2 \times 2) \rightarrow$ ResBlock	64	128
Down 3	MaxPool $(2 \times 2) \rightarrow$ ResBlock	128	256
Bottleneck	MaxPool $(2 \times 2) \rightarrow$ ResBlock	256	512
	Decoder Path		
Up 1	Upsample $(2\times) \rightarrow$ Concat w/ Down 3 $\rightarrow$ ResBlock $\rightarrow$ CBAM	$512 + 256$	256
Up 2	Upsample $(2\times) \rightarrow$ Concat w/ Down 2 $\rightarrow$ ResBlock $\rightarrow$ CBAM	$256 + 128$	128
Up 3	Upsample $(2\times) \rightarrow$ Concat w/ Down 1 $\rightarrow$ ResBlock $\rightarrow$ CBAM	$128 + 64$	64
Up 4	Upsample $(2\times) \rightarrow$ Concat w/ Stem $\rightarrow$ ResBlock $\rightarrow$ CBAM	$64 + 32$	32

B.2 TASK-SPECIFIC HEADS

While the backbone is shared, the final projection layers differ for the embedder and detector tasks:

- Embedder Head (emb_θ): The output of the backbone (32 channels) is passed through a final 1×1 convolution to map back to the input channel space ($C = 3$). The result is added to the original image as a residual.
- Detector Head (det_ϕ): The output of the backbone (32 channels) is passed through a 1×1 convolution to produce a single-channel logit map ($C = 1$), which is processed by a Sigmoid function during inference to generate the tampering probability mask.

C WATERMARK VISUZALIZATION

D LOSSES

Fidelity loss. The fidelity term encourages the watermarked frame $\mathcal{Z}$ to remain visually close to the original frame $\mathcal{X}$ by combining pixel-wise, signal-quality, and perceptual criteria:

$$\mathcal{L}_{\text{fid}} = \mathrm{MSE}(\mathcal{Z}, \mathcal{X}) + \big(1 - \mathrm{SSIM}(\mathcal{Z}_\downarrow, \mathcal{X}_\downarrow)\big) + \mathrm{LPIPS}(\mathcal{Z}_\downarrow, \mathcal{X}_\downarrow), \tag{5}$$

where $\mathcal{Z}_\downarrow = \mathrm{Down}_{H' \times W'}(\mathcal{Z})$ and $\mathcal{X}_\downarrow = \mathrm{Down}_{H' \times W'}(\mathcal{X})$ denote bilinearly downsampled versions at resolution $H' \times W'$. We use SSIM as the loss $1 - \mathrm{SSIM}(\cdot, \cdot)$ since higher SSIM indicates better structural similarity. Here, MSE denotes mean squared error, SSIM the structural similarity index, and LPIPS the learned perceptual image patch similarity.

Localization loss. The localization term supervises the predicted tampering mask $M \in \{0, 1\}^{H \times W}$ using the ground-truth mask $M^{\text{gt}} \in \{0, 1\}^{H \times W}$:

$$\mathcal{L}_{\text{loc}} = \mathrm{BCE}(M, M^{\text{gt}}) + \big(1 - \mathrm{Dice}(M, M^{\text{gt}})\big), \tag{6}$$

where BCE denotes binary cross-entropy and Dice is the Dice similarity coefficient (used as $1 - \mathrm{Dice}$ to form a loss). This combination balances pixel-wise supervision (BCE) with overlap-based supervision (Dice), which is beneficial under class imbalance between tampered and pristine regions.

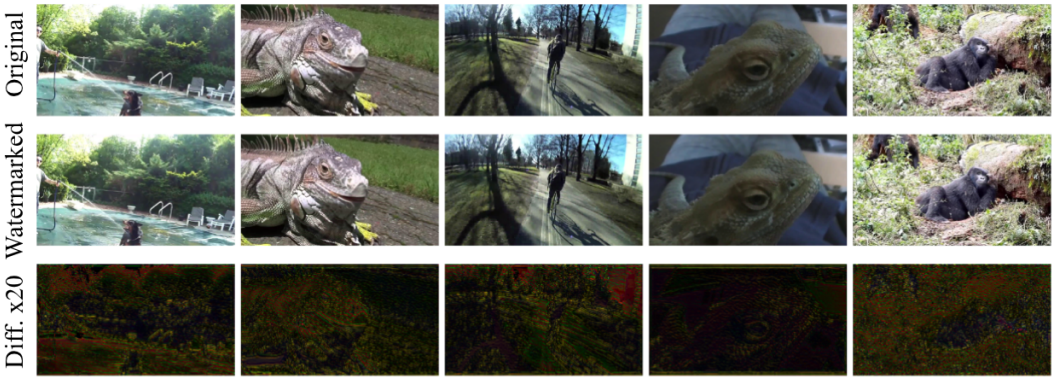

Figure 4: Visualization of the watermarking process across five test videos. The first row displays the original frames. The second row shows the watermarked frames, which remain visually indistinguishable from the originals. The third row reveals the amplified residual difference (watermark signal) added to the video frames.