Supplementary Material

Hanyi Wang[1] Han Fang[2,*] Shi-Lin Wang[1] Ee-Chien Chang[2]
[1]Shanghai Jiao Tong University [2]National University of Singapore
why_820@sjtu.edu.cn, fanghan@nus.edu.sg, wsl@sjtu.edu.cn, changec@comp.nus.edu.sg

1. Analysis of Regenration-based Optimization strategy

To demonstrate why this optimization can yield better inversion results, we analyze a single step of the forward and inversion process. Specifically, we first generate x_{t-1} from x_t and then apply DDIM inversion to approximately reconstruct x'_t from x_{t-1}. By combining Eq. 4 and Eq. 6 of the submitted paper, we derive the difference between the approximate reverse representation x'_t and the ground truth x_t as follows:

$$\begin{aligned} x'_t - x_t = \frac{\varphi_t}{\gamma_t}\{&w[\epsilon_\theta(x_t, t, C) - \epsilon_\theta(x_t, t, \emptyset)] \\ &+ \epsilon_\theta(x_t, t, \emptyset) - \epsilon_\theta(x_{t-1}, t, \emptyset)\}. \end{aligned} \tag{1}$$

We further denote the optimized latent representation as x^*_t, aiming to bring it closer to the original x_t. Based on the optimization objective, x^*_t is constrained to ensure that it can generate x_{t-1} in a single step:

$$x_{t-1} = \gamma_t x^*_t + \varphi_t \epsilon_\theta(x^*_t, t, \emptyset). \tag{2}$$

Thus, the difference between the optimized reverse representation x^*_t and the ground truth x_t is given by:

$$\begin{aligned} x^*_t - x_t = \frac{\varphi_t}{\gamma_t}\{&w[\epsilon_\theta(x_t, t, C) - \epsilon_\theta(x_t, t, \emptyset)] \\ &+ \epsilon_\theta(x_t, t, \emptyset) - \epsilon_\theta(x^*_t, t, \emptyset)\}. \end{aligned} \tag{3}$$

By comparing Eq. 1 and Eq. 3, we observe that the relationship between $\|x^*_t - x_t\|$ and $\|x'_t - x_t\|$ can be analyzed through the magnitudes of $\|\epsilon_\theta(x_t, t, \emptyset) - \epsilon_\theta(x^*_t, t, \emptyset)\|$ and $\|\epsilon_\theta(x_t, t, \emptyset) - \epsilon_\theta(x_{t-1}, t, \emptyset)\|$.

Furthermore, we hypothesize that $\epsilon_\theta(x, t, \emptyset)$ exhibits local linearity within a small neighborhood around x_t. This assumption follows from the discrete nature of both computation and the underlying noise schedule, as suggested in prior works on first-order ODE solvers [7]. Similar linearization assumptions have been commonly adopted in diffusion-based generative models [4, 6], where the learned noise prediction function is observed to behave smoothly in local regions. Under this assumption, the comparison can be further transformed into analyzing the relative magnitudes of $\|x_t - x^*_t\|$ and $\|x_t - x_{t-1}\|$.

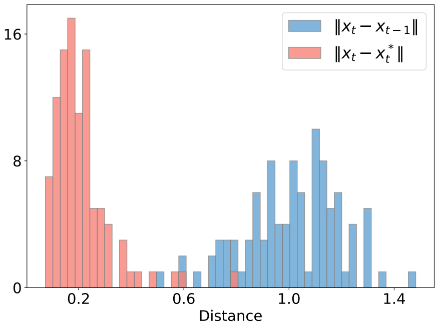

Figure 1. Histogram between $\|x_t - x^*_t\|$ and $\|x_t - x_{t-1}\|$

Based on the above analysis, we empirically conducted statistical experiments to measure the Euclidean distances between x_t and its approximations (x^*_t and x'_t) across different inversion steps and datasets. As consistently demonstrated in Fig. 1, the results show that $\|x_t - x^*_t\|$ is significantly smaller than $\|x_t - x_{t-1}\|$, supporting our claim that the optimization process effectively reduces inversion error and improves reconstruction accuracy.

2. The implementation details of compared baseline methods

We compare ROAR with eight state-of-the-art generative model watermarking frameworks, categorized into three groups. For all methods, we utilize their officially released open-source implementations and configure the watermark parameters based on their respective specifications.

(1) Image watermarking-based methods, including DwtDct[2], DwtDctSvd [2] and RivaGAN [10]. We utilize the implementation provided in an open-source repository[1]. The watermark capacity is set to 256 bits for DwtDct and DwtDctSvd, while RivaGAN is limited to its maximum ca-

[*] Corresponding author.

[1] https://github.com/ShieldMnt/invisible-watermark

pacity of 32 bits.

(2) Fine-tuning-based methods, including Stable Signature [3] and LaWa [5]. We adopt the open-source code of Stable Signature[2], setting the watermark capacity to 48 bits. Similarly, we use the publicly available code for LaWa[3], also setting the capacity to 48 bits.

(3) Inversion-based methods, Tree-Ring [8], RingID [1], and Gaussian Shading (GS) [9]. For Tree-Ring[4], the detailed parameters are $w\_channel$=0, $w\_pattern$='ring', w_radius=10, with only 1-bit watermark. For RingID[5], the detailed parametes are $ring\_width$=1, $quantization\_levels$=2, $ring\_value\_range$=64. For GS[6], the detailed parameters are f_c=1, f_hw=8, l=1, with an watermark capacity of 256 bits.

3. TPR calculation with multi-bits methods

To calculate the TPR of multi-bits methods, we adopt the same TPR calculation method proposed by GS[9], which can be described as follows: Assume the watermark to be embedded is $w \in \{0,1\}^k$ where k is the length. For the image to be detected x, we can extract the watermark w_x from the x and calculate the bit accuracy of w_x, noted as $Acc(w_x, w)$. Then a threshold τ is applied to make a decision: if

$$Acc(w_x, w) \geq \tau,$$

x is regarded as watermarked. In this form, τ is set based on a required estimated false positive rate (FPR), which is defined as the probability that $Acc(w_{x'}, w)$ of a non-watermarked image x' exceeds the threshold τ. Such a probability can be further calculated with regularized incomplete beta function $B_x(a;b)$[3]:

$$\begin{aligned}\mathrm{FPR}(\tau) &= \mathcal{P}\left(Acc\left(w_{x'}, w\right) > \tau\right) = \frac{1}{2^k} \sum_{i=\tau+1}^{k} \binom{k}{i} \\ &= B_{1/2}(\tau+1, k-\tau).\end{aligned}$$

Detail analysis of the TPR calculation can be found in [9].

Shortly, to calculate the TPR, we first set an FPR (e.g. 10^{-10} in this paper). Then according to the set FPR, we can determine a threshold τ with the embedded watermark length k. The image with extraction bit accuracy larger than τ is regarded as watermarked.

4. Details of Comparison Experiments

We provide detailed results of true positive results and bit accuracy results for each distortion in Table 1 and Tab. 2. The distortions we tested are: JPEG, QF=25; 60% area Random Crop (RandCr); 80% area Random Drop (RandDr); Gaussian Blur $r = 4$ (GauBlur); Median Filter, $k = 7$ (MedFilter); Gaussian Noise, $\mu = 0, \sigma = 0.05$ (GauNoise); Salt and Pepper Noise, $p = 0.05$ (S&P Noise); 25% Resize and restore (Resize); Brightness, $factor = 6$.

From the observed results, most of the compared methods struggle to maintain high extraction accuracy and true positive detection rates under distortion scenarios. In contrast, under detection conditions, our method achieves an average TPR exceeding 0.999, outperforming Tree-Ring, RingID, and GS by 14.15%, 10.35% and 8.2%, respectively. In the traceability scenario, our approach improves the average bit accuracy compared to GS by 9.98%, demonstrating its robustness and effectiveness.

5. Details of Adaptive Attack Experiments

In Section 5.3, we discuss the adaptive attack on the proposed methods. We believe that a successful attack should meet two requirements: 1). the visual consistency where the attacked images should be similar to the original image. 2). the watermark cannot be detected in the attacked images. We adopt two types of adaptive attacks, including reconstruction attack and purification attack. Notably, experimental results from Table 2 in the submitted paper demonstrate that our method exhibits high robustness against reconstruction attacks. However, it faces challenges when subjected to a potential purification attack, where the attacker introduces random noise into the watermarked image and then applies a diffusion process to remove both the noise and the watermark. Our experiments show that the proposed method is robust to such attacks with s = 0.05 to 0.3, but got a performance decrease in the face of stronger distortion. Here, we give the visual results of the purification attack, as shown in Fig. 2. It can be seen that when s is small (e.g. σ = 0.05), the attacked image maintains a high similarity to the original image. However, with the increase of s, the appearance of the attacked images changes a lot, especially when s = 0.7, both the detail and the structure of the image change a lot. Although such a process can erase the watermark, it fails to meet the requirements of visual consistency. Besides, from Table 2 in submitted papers we can see that, even under s = 0.3, the watermark can still be detected/extracted, which indicates the certain robustness of the proposed scheme against purification attack.

6. Optimization Process of the Regeneration-based Optimization Mechanism

Our proposed Regeneration-based Optimization Mechanism begins by initializing the optimization process with the latent representation obtained from DDIM inversion. This latent representation is iteratively refined over 20 iterations

[2] https://github.com/facebookresearch/stable_signature
[3] LaWa Official Code
[4] https://github.com/YuxinWenRick/tree-ring-watermark
[5] https://github.com/showlab/RingID
[6] https://github.com/bsmhmmlf/Gaussian-Shading

Methods	DwtDct	DwtDctSVD	RivaGAN	Stable Signature	LaWa	Tree-Ring	Tree-Ring-ROAR	RingID	RingID-ROAR	GS	GS-ROAR
JPEG	0.000/0.000	0.000/0.000	0.000/0.000	0.000/0.000	0.130/0.120	0.830/0.840	1.000/1.000	0.870/0.930	1.000/1.000	0.930/0.947	1.000/1.000
RandCr	0.890/0.870	1.000/1.000	0.680/0.680	0.870/0.860	0.000/0.020	0.970/0.973	1.000/1.000	0.965/0.967	1.000/1.000	0.982/0.981	1.000/1.000
RandDr	0.000/0.000	0.000/0.000	0.910/0.970	0.940/1.000	1.000/1.000	0.940/1.000	1.000/1.000	0.940/0.944	1.000/1.000	0.975/0.986	1.000/1.000
GauBlur	0.000/0.000	0.080/0.050	0.000/0.000	0.000/0.000	0.000/0.000	0.740/0.780	1.000/1.000	0.801/0.802	1.000/1.000	0.820/0.835	1.000/1.000
MedFilter	0.000/0.000	0.750/0.900	0.160/0.140	0.000/0.000	0.020/0.030	0.840/0.863	1.000/1.000	0.802/0.826	1.000/1.000	0.810/0.828	1.000/1.000
GauNoise	0.000/0.000	0.000/0.000	0.040/0.110	0.000/0.000	0.150/0.160	0.710/0.760	1.000/1.000	0.870/0.883	1.000/1.000	0.880/0.882	1.000/1.000
S&P Noise	0.000/0.000	0.000/0.000	0.010/0.010	0.000/0.000	0.050/0.050	0.730/0.789	1.000/1.000	0.862/0.879	1.000/1.000	0.860/0.874	1.000/1.000
Resize	0.000/0.000	0.985/0.983	0.850/0.887	0.000/0.000	0.000/0.000	0.940/1.000	1.000/1.000	0.940/0.952	1.000/1.000	0.972/0.983	1.000/1.000
Brightness	0.110/0.100	0.110/0.080	0.450/0.510	0.700/0.610	0.970/0.960	0.850/0.900	1.000/1.000	0.900/0.917	1.000/1.000	0.980/1.000	1.000/1.000
Avg.	0.111/0.108	0.325/0.335	0.344/0.367	0.279/0.274	0.258/0.264	0.839/0.878	1.000/1.000	0.894/0.899	1.000/1.000	0.912/0.924	1.000/1.000

Table 1. Comparison of different watermarking methods under various perturbations in terms of TPR for SD V1.4/2.1.

Methods	DwtDct	DwtDctSVD	RivaGAN	Stable Signature	LaWa	GS	GS-ROAR
JPEG	0.4964/0.5030	0.4579/0.4537	0.5894/0.6006	0.5690/0.5710	0.7929/0.7910	0.9506/0.9563	0.9907/0.9900
RandCr	0.8650/0.8344	0.9988/0.9990	0.8931/0.9172	0.9404/0.9552	0.7485/0.7498	0.9907/0.9932	0.9989/0.9990
RandDr	0.5126/0.5061	0.4998/0.4996	0.9678/0.9906	0.9627/0.9938	1.0000/1.0000	0.9872/0.9896	0.9992/0.9993
GauBlur	0.4991/0.5067	0.5675/0.5608	0.5353/0.5309	0.3992/0.3967	0.5800/0.5827	0.7314/0.7347	0.9629/0.9634
MedFilter	0.4998/0.5111	0.7682/0.7791	0.7388/0.7469	0.4619/0.4704	0.7765/0.7763	0.7079/0.7134	0.9782/0.9784
GauNoise	0.4786/0.4702	0.5266/0.5285	0.7113/0.7428	0.5352/0.5419	0.7804/0.7902	0.8105/0.8275	0.9922/0.9909
S&P Noise	0.5033/0.5038	0.5059/0.5119	0.7016/0.7128	0.5365/0.5346	0.7469/0.7594	0.8277/0.8301	0.9924/0.9910
Resize	0.5067/0.5135	0.8743/0.8630	0.9602/0.9733	0.5067/0.5163	0.5567/0.5606	0.9796/0.9987	0.98180.9993
Brightness	0.5031/0.5091	0.5258/0.5124	0.8378/0.8328	0.9017/0.8835	0.9900/0.9850	0.9763/0.9991	0.9926/0.9912
Avg.	0.5405/0.5398	0.6361/0.6342	0.7706/0.7831	0.6459/0.6515	0.7747/0.7772	0.8847/0.8936	0.9887/0.9892

Table 2. Comparison of different watermarking methods under various perturbations in terms of bit accuracy for SD V1.4/2.1.

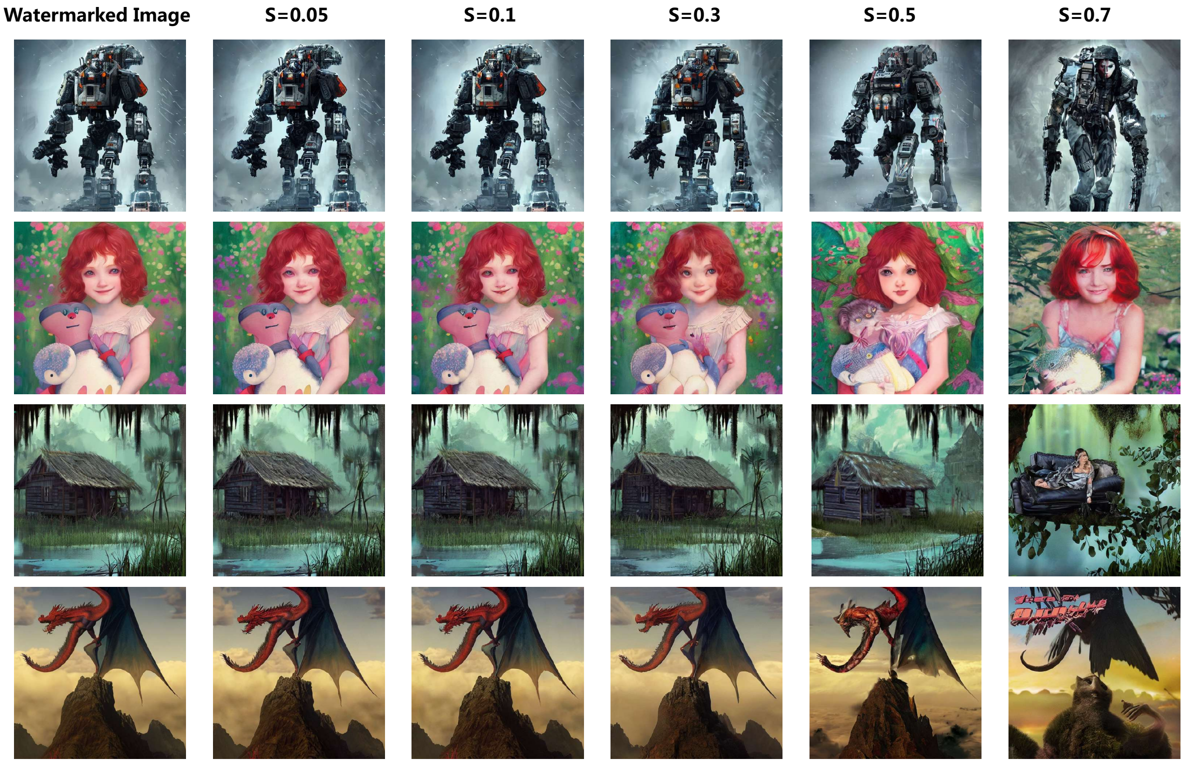

Figure 2. Visualization of purification attack.

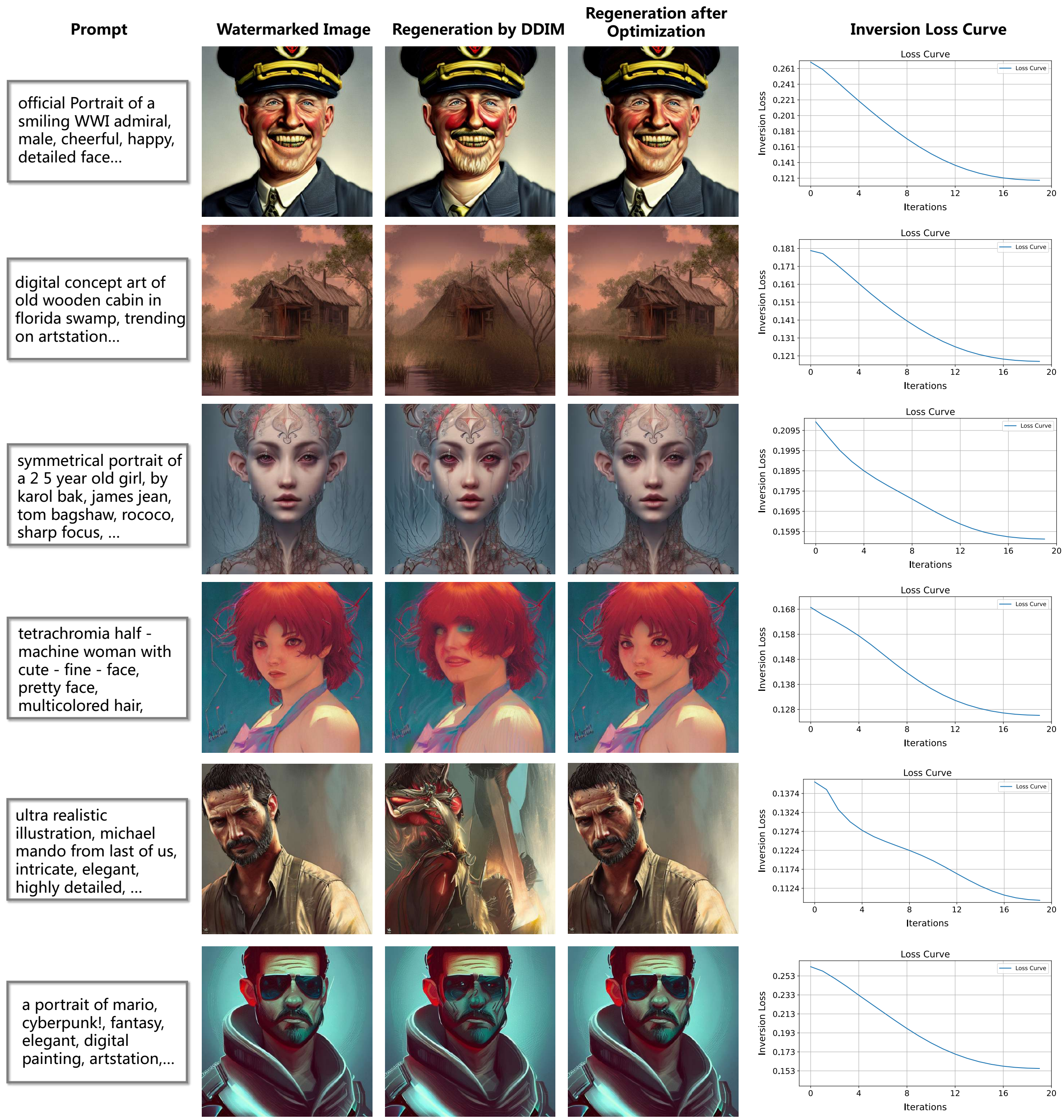

Figure 3. Visualization of the optimization process of Tree-Ring. (a) The original watermarked image. (b) The image reconstructed from the initial latent representation obtained via DDIM inversion. (c) The image reconstructed from the optimized latent representation (d) The inversion loss variation during the optimization process.

to enhance the quality of the regenerated image. In Figure 3, 4 and 5, we illustrate the entire optimization process of Tree-Ring, RingID and Gaussian Shading, respectively, including (1) the original watermarked image, (2) the image generated from the initial latent representation obtained through DDIM inversion, (3) the image generated from the optimized latent representation after 20 iterations, and (4) the corresponding loss variation throughout the optimiza-

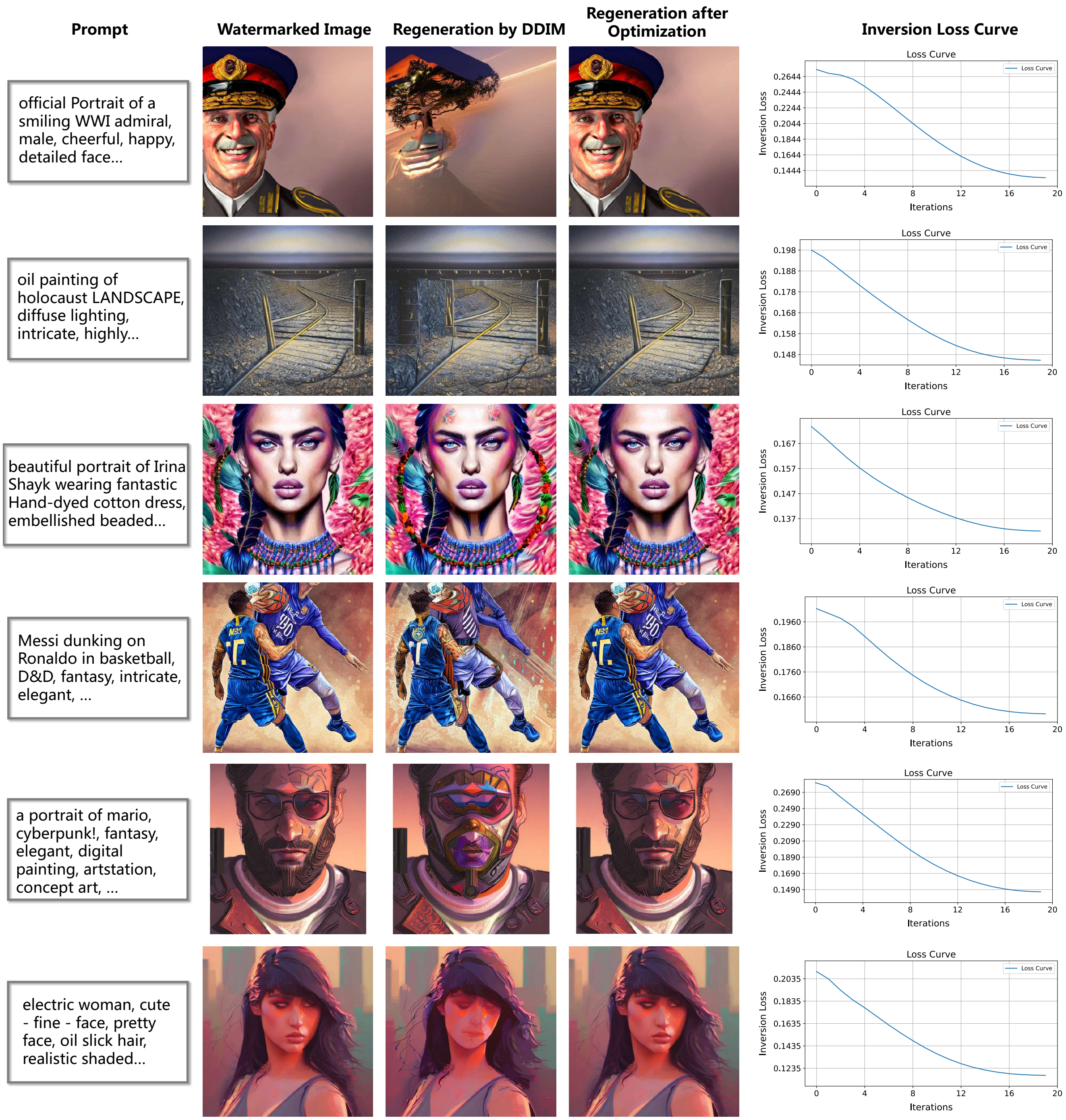

Figure 4. Visualization of the optimization process of RingID. (a) The original watermarked image. (b) The image reconstructed from the initial latent representation obtained via DDIM inversion. (c) The image reconstructed from the optimized latent representation (d) The inversion loss variation during the optimization process.

tion process. This visualization demonstrates that the optimized latent representation becomes more accurate and better aligned with the original watermarked image.

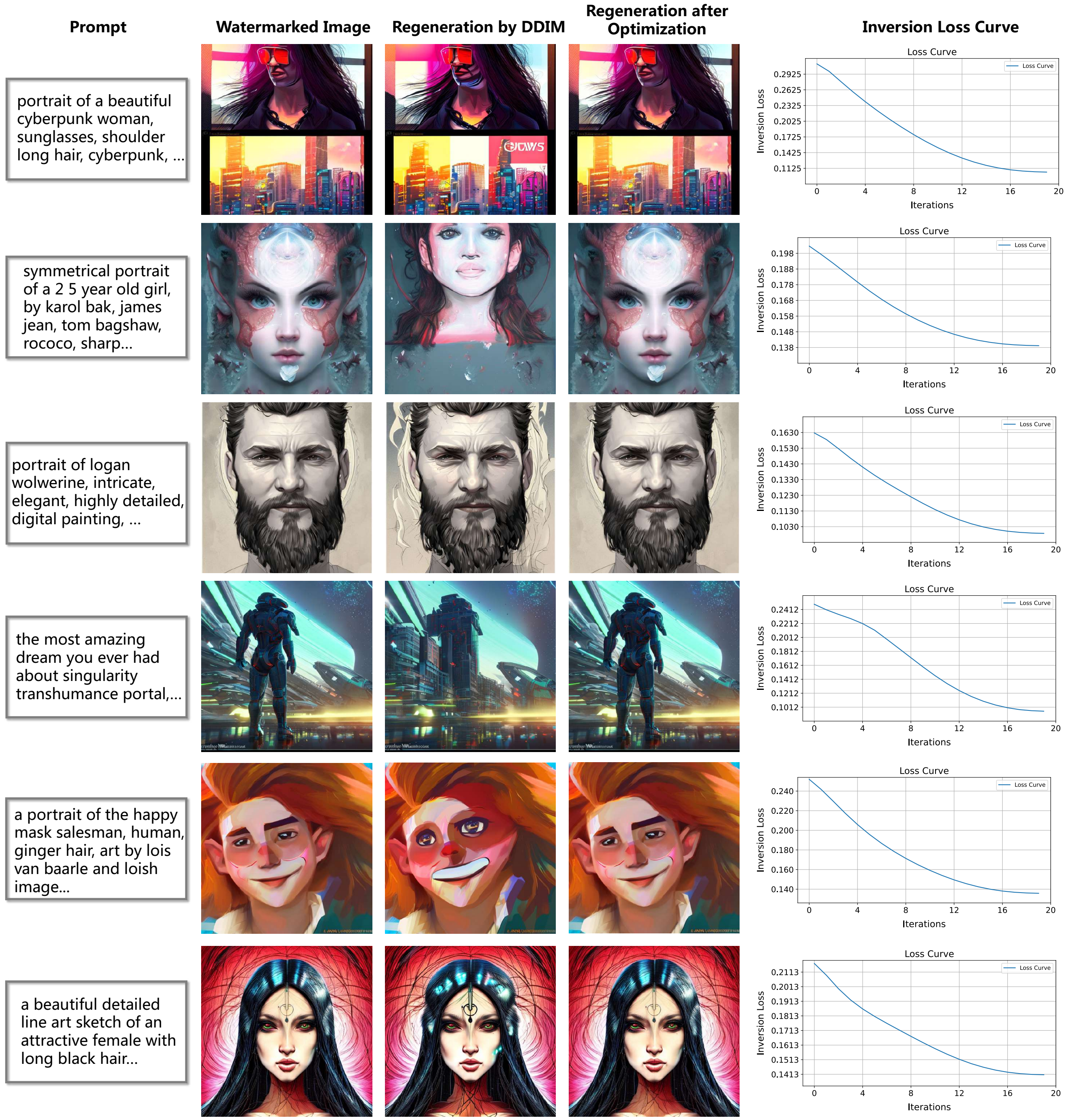

Figure 5. Visualization of the optimization process of GS. (a) The original watermarked image. (b) The image reconstructed from the initial latent representation obtained via DDIM inversion. (c) The image reconstructed from the optimized latent representation (d) The inversion loss variation during the optimization process.

References

[1] Hai Ci, Pei Yang, Yiren Song, and Mike Zheng Shou. Ringid: Rethinking tree-ring watermarking for enhanced multi-key identification. In *European Conference on Computer Vision*, pages 338–354. Springer, 2025. 2

[2] Ingemar Cox, Matthew Miller, Jeffrey Bloom, Jessica Fridrich, and Ton Kalker. *Digital watermarking and steganography*. Morgan kaufmann, 2007. 1

[3] 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*, pages 22466–22477, 2023. 2

[4] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021. 1

[5] Ahmad Rezaei, Mohammad Akbari, Saeed Ranjbar Alvar, Arezou Fatemi, and Yong Zhang. Lawa: Using latent space for in-generation image watermarking. In *European Conference on Computer Vision*, pages 118–136. Springer, 2025. 2

[6] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 1

[7] Gerhard Wanner and Ernst Hairer. *Solving ordinary differential equations II*. Springer Berlin Heidelberg New York, 1996. 1

[8] 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. 2

[9] Zijin Yang, Kai Zeng, Kejiang Chen, Han Fang, Weiming Zhang, and Nenghai Yu. Gaussian shading: Provable performance-lossless image watermarking for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12162–12171, 2024. 2

[10] Kevin Alex Zhang, Lei Xu, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Robust invisible video watermarking with attention. *arXiv preprint arXiv:1909.01285*, 2019. 1