ARE SEMANTIC WATERMARKS FOR DIFFUSION MODELS RESILIENT TO LAYOUT CONTROL?

Denis Lukovnikov, Andreas Müller, Jonas Thietke, Erwin Quiring, Asja Fischer
Ruhr University Bochum, Germany
denis.lukovnikov@rub.de

ABSTRACT

Semantic watermarking methods embed information into generated images by modifying the initial latent noise, subtly modifying the output images. However, the widespread use of layout control techniques, such as ControlNets, raises questions about the applicability of semantic watermarking with layout control. After all, if semantic watermarks are really realized as meaningful changes in images (such as its layout), external layout specifications (e.g. through edge maps), could destroy the watermark information during denoising. This work empirically evaluates two semantic watermarking approaches—Tree-Ring Watermarking and Gaussian Shading—under various ControlNet-guided generation settings. Our results show that while ControlNets can slightly degrade watermark strength, both watermarking approaches remain largely detectable, demonstrating the potential viability of semantic watermarks even under strong layout constraints.

1 INTRODUCTION

The rapid advancement of generative models for image synthesis has revolutionized creative workflows and content production. Latent diffusion models (LDMs) like Stable Diffusion (Rombach et al., 2022) and FLUX.1 can generate high-quality images from textual prompts. ControlNets (Zhang et al., 2023; Zhao et al., 2024a) and similar works (Mou et al., 2024) enhance this capability by providing more control over the layout of the generated images. They are a class of neural network architectures designed to impose fine-grained control over generative models by conditioning outputs on auxiliary inputs (e.g., edge maps, depth maps, or poses). This enables users to guide image generation with high precision.

However, as generative models proliferate, concerns about intellectual property, content authenticity, and misuse have intensified. This has spurred interest in watermarking methods. They enable a service provider—offering an API to a (private) generative model—to hide information into each generated image without affecting its overall quality. The embedded information can indicate that the image is AI-generated, or can identify the user or service provider that generated the image. Technically, this can be achieved in various ways for LDMs, such as classic post-hoc watermarking (Cox et al., 2002) or LDM decoder fine-tuning (Fernandez et al., 2023). In this work, we focus on recently proposed *semantic watermarks* that rely on the inversion of the denoising process in the diffusion model (Wen et al., 2023; Yang et al., 2024; Ci et al., 2024; Gunn et al., 2024). They modify the initial latent noise to incorporate a watermark pattern during generation, which can be then retrieved through the inversion of the denoising process. Hence, semantic watermarks are diffused across the image and realized, for example, in object details and their arrangements (see Figure 5 in the Supplementary Material). These watermarks thus leverage the fact that there are numerous ways to generate an image that conforms to user specifications, in contrast to earlier watermarking methods that add imperceptible noise patterns on top of the image.

In this work, we examine whether semantic watermarks are compatible with layout control (in particular, ControlNets), which is important for the practical deployment of watermarking. At first glance, they should not be compatible. Semantic watermarks are assumed to be realized as part of the image layout and edges (Ci et al., 2024; Saberi et al., 2024). As ControlNets provide users with fine-grained control over image layout, watermarking information might be therefore erased during the denoising process when it is guided by a ControlNet. We present an extensive empirical study

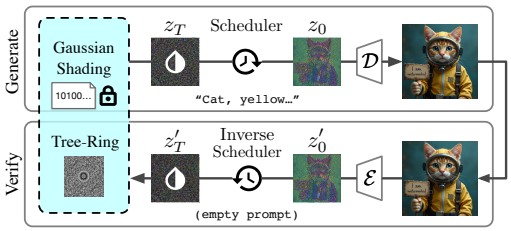

Figure 1: Illustration of inversion-based semantic watermarking in the text-to-image setting.

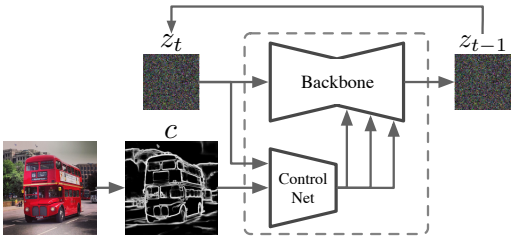

Figure 2: Illustration of how ControlNets work.

that examines two representative inversion-based semantic watermarks, Tree-Ring watermarking (TRW) (Wen et al., 2023) and Gaussian Shading (GS) (Yang et al., 2024), and studies their effectiveness for different types of control signals, under different common perturbations.

2 Background

2.1 Semantic Watermarking

Semantic watermarking methods modify the distribution of the generator, such that the watermark is not encoded as an imperceptible noise pattern applied on a clean image, but rather the generation process is influenced such that the generated images implement the watermark through more meaningful patterns. Compared to imperceptible watermarks, semantic watermarks are more resilient to certain adversarial attacks such as regeneration using autoencoders and diffusion models (Zhao et al., 2024b; An et al., 2024), as well as a range of common image transformations.

Figure 1 illustrates the technical concept. The core idea is to sample initial latents z_T not from $\mathcal{N}(O, I)$, but from a modified distribution. The different watermarking approaches differ in how they change the initial latent. For example, Tree-Ring (Wen et al., 2023) modifies z_T such that its frequency spectrum carries visible concentric tree-ring patterns. Gaussian Shading (Yang et al., 2024) employs cryptography to generate pseudorandom ciphertext that drives the sampling of z_T. Given the watermarked z_T, the image generation then simply follows the standard diffusion model process. Finally, verification is done by running inverse sampling to recover an approximation of z_T and verifying the presence of the watermarking pattern in the approximated z'_T.

2.2 ControlNet

ControlNets (Zhang et al., 2023) are an extension of diffusion models that allow for fine-grained control over the generation process by conditioning on structured guidance inputs, such as edge maps, depth maps, or segmentation masks. Figure 2 illustrates the process. A ControlNet consists of an auxiliary neural network that is trained to process an external control signal, presented as an image. The control signals can be derived from various sources, such as edges extracted from an image, human pose estimation, or depth information, allowing for greater flexibility in controlling the generated images. In Figure 2, for instance, the control signal is an edge map of the bus.

In general, a ControlNet is initialized from a (partial) copy of the backbone diffusion model. The ControlNet's latent features from different layers are injected into the backbone using zero-convolutions. During training, the backbone parameters are frozen and the ControlNet is fine-tuned to steer the denoising process towards a solution that adheres to the given control signal. During generation, the control signal has an effect on the backbone's internal features throughout the entire denoising process. It is unclear if this interferes with the expression of the semantic watermark that is injected into the initial latent z_T.

3 Semantic Watermarking and Layout Control

In this work, we aim to investigate how inversion-based semantic watermarks behave in combination with layout control throughout the denoising process. Intuitively, we might expect that layout con-

	Tree-Ring Watermarking					Gaussian Shading				
	None	Canny	HED	Dpth	Nrml	None	Canny	HED	Dpth	Nrml
Clean	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
JPEG	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
G.N.	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
SP.N.	0.992	0.996	1.000	0.996	0.992	1.000	1.000	1.000	1.000	0.996
G.N.+JPEG	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

Table 1: Watermark detection ratio for Tree-Ring Watermarking and Gaussian Shading with different ControlNets (columns) and different perturbations applied after generation (rows).

trol should interfere with semantic watermarking. First, it is unclear to what degree the watermarks are expressed in terms of details in the image layout or edges. When comparing non-watermarked images with watermarked ones using Tree-Ring watermarking, it appears that the layout of the image is affected, suggesting that the watermarks are realized (at least partly) through image layout and edges, which is also assumed by some prior work (Ci et al., 2024). If the watermarks are entirely dictated by edge information, then an edge-conditioned ControlNet could completely destroy the watermark information. Second, during watermark verification, an inversion of the image sampling process is required. However, the inversion needs to be performed without any conditioning information that has guided the denoising process, such as prompt or control signal. This is because in a real watermarking scenario, the service provider cannot track all conditioning information of all ever generated images. Thus, the inversion must proceed with incomplete information and only with the backbone model. If the composition of the backbone and a ControlNet is sufficiently different to the original backbone, inversion using just the backbone can lead to a different trajectory and may not recover the watermark. In the experiments reported below, we extract control signals (e.g., edge maps) from unwatermarked real images and use a watermarked z_T to start the generation process.

In addition to ControlNets, we also investigate a more extreme form of image control that tries to regenerate the original image using a conditioned diffusion model. Where a normal diffusion model takes z_t as input, *inpainting* versions of models have been developed that in addition to the generated z_t also take a reference image's latent representation $z_0^{(\mathrm{ref})}$ as well as an inpainting mask m. When an empty mask is specified (indicating that nothing has to be inpainted), these models can be used for creating a close copy of the original image that has the same layout, edges and color but may differ slightly in insignificant details. In our experiments, we use an unwatermarked real image as reference image to be copied, while starting decoding from a watermarked initial latent z_T[1].

4 Experiments

In the following, we empirically test if semantic watermarks are usable with layout control. To this end, we perform a series of experiments to see if the watermark is detectable, remains robust under perturbations, and preserves the quality of the generated images.

4.1 Experimental Setup

We use the following control methods: Canny edge, HED, depth map, and normal map[2]. Throughout the evaluation, we use a subset of 500 images of a resolution of at least 512×512 and corresponding captions from the MS-COCO 2017 Lin et al. (2014) training dataset. We then use the captions to generate images of size 512×512 using Stable Diffusion 1.5[3] with the respective ControlNets. For the image regeneration experiments, we used SD2's inpainting variant. During generation, to create unwatermarked images (*clean*), we use randomly sampled latents z_t. To get watermarked images, we apply one of the semantic watermarking methods, Tree-Ring Watermarking (TRW) (Wen et al.,

[1]This is somewhat similar to the watermark removal approach presented in Liu et al. (2024), with the key difference that we investigate opposite phenomena and look at resilience of watermarked z_T's whereas they started from clean z_T and used regeneration with specially trained components for watermark removal.

[2]Huggingface ControlNet Model cards: Canny, HED Boundary, Depth, Normal Map

[3]Stable Diffusion 1.5 Huggingface model card

		Canny	HED	Depth	Normal	No Control
Clean	TRW (mean)	8.36×10^{-14}	6.79×10^{-15}	1.43×10^{-13}	2.79×10^{-15}	3.34×10^{-18}
	TRW (median)	3.19×10^{-24}	1.15×10^{-24}	5.85×10^{-27}	3.77×10^{-30}	4.57×10^{-33}
	GS (mean)	1.000	1.000	1.000	1.000	1.000
JPEG	TRW (mean)	1.23×10^{-8}	8.12×10^{-10}	1.55×10^{-8}	2.36×10^{-9}	4.51×10^{-10}
	TRW (median)	7.02×10^{-18}	3.27×10^{-19}	1.25×10^{-19}	1.65×10^{-18}	3.49×10^{-23}
	GS (mean)	1.000	1.000	1.000	1.000	1.000
G.N.	TRW (mean)	1.95×10^{-6}	8.18×10^{-7}	4.42×10^{-6}	2.17×10^{-5}	9.56×10^{-6}
	TRW (median)	1.59×10^{-11}	1.86×10^{-12}	1.00×10^{-11}	2.21×10^{-10}	6.64×10^{-12}
	GS (mean)	0.997	0.997	0.996	0.991	0.996
SP.N.	TRW (mean)	6.28×10^{-4}	2.33×10^{-4}	3.78×10^{-4}	7.91×10^{-4}	6.76×10^{-4}
	TRW (median)	3.70×10^{-6}	3.50×10^{-7}	2.22×10^{-6}	6.31×10^{-6}	1.87×10^{-6}
	GS (mean)	0.935	0.949	0.940	0.906	0.933
G.N. + JPEG	TRW (mean)	2.57×10^{-5}	1.71×10^{-6}	7.29×10^{-6}	1.88×10^{-5}	1.24×10^{-5}
	TRW (median)	5.30×10^{-11}	5.88×10^{-12}	4.54×10^{-11}	5.62×10^{-10}	1.36×10^{-11}
	GS (mean)	0.995	0.996	0.994	0.989	0.995

Table 2: Watermarking performance for different ControlNets. For Tree-Ring Watermarking (TRW), the average p-values are reported and for Gaussian Shading (GS), the average bit accuracies. Note that when computing the averages, the two most extreme values were discarded.

2023) and Gaussian Shading (GS) (Yang et al., 2024). For TRW, we insert a ring pattern into the same latents by using the provided implementation with default parameters and the *Rings* option[4]. For GS, each image is generated from a novel latent z_T which is drawn by performing a specific sampling from an encrypted bit string that varies for each image. We choose default options, i.e., a message capacity of 256 bits and $\times 64$ message replication[5].

TRW is evaluated in a detection setup, while the threshold for GS is set for an identification scenario with 100k users (each having a distinct random watermark message). As metrics, we measure the watermark detection ratio, and also measure the p-values for TRW and bit accuracies for GS, as reported in their respective publications. The p-value indicates how likely it is to observe the tree-ring pattern by random chance. Bit accuracy indicates how much of the original message is reconstructed correctly. Put simply, we strive for a smaller p-value in Tree-Ring and higher bit accuracy in Gaussian Shading. Finally, we also report AUROC and TPR against images generated under the same conditions but from unwatermarked z_T.

In line with previous work, we also test various common image perturbations. These perturbations include JPEG compression with quality factor 82 ("JPEG" in tables), Gaussian Noise with $\sigma = 0.1$ ("G.N."), Salt-and-Pepper noise with 5% probability ("SP.N."), as well as the combination of Gaussian Noise with $\sigma = 0.1$ and JPEG compression with quality factor 82 ("G.N.+JPEG"). In preliminary experiments, we saw that even small rotation and crop-and-scale transformations can make watermarks undetectable and for this reason do not include this in our evaluation. This is consistent with earlier work (Müller et al. (2025); An et al. (2024)).

4.2 Results

Table 1 shows the detection success for Tree-Ring and Gaussian Shading for various ControlNets. The detection is still successful under all tested ControlNets, even with more challenging perturbations, such as Salt-and-Pepper noise.

Table 2 presents the p-values and bit accuracies for TRW and GS, respectively. As averaged p-values can suffer from extreme outliers, we also provide the median p-values. When comparing the different ControlNets to the baseline case without control (last column of Table 2), it is visible that the use of ControlNets negatively affects the p-values for TRW (as visible in the first rows of Table 2). Still, the values remain well below the detection threshold (~ 0.02). For GS, the bit accuracies are on average 100% or close, and remain similar to the case without using ControlNets.

[4]Tree-Ring Github repository

[5]Gaussian Shading Github repository

	Mean (median) p-val./bit acc.	Det. Acc.	AUROC	TPR
TRW	6.24×10^{-2} (1.59×10^{-2})	0.587	0.940	0.535
GS	0.826	0.985	1.00	1.00

Table 3: Evaluation results for image regeneration using an inpainting model (SD2).

Figure 3: Example of images with semantic watermarks generated with canny edge control. The caption used as prompt is *"A living room with a cream colored couch"*. The clean image is generated from a randomly drawn z_T. The TRW image is generated from the same z_T with the Tree-Ring pattern inserted. The TRW image is verified as watermarked with a p-value of 4.3×10^{-42}. The GS image is verified as watermarked with a bit accuracy of 1.0.

Tables 4 and 5 in the Supplementary Material provide further results by reporting the AUROC and TPR for TRW and GS with different control signals. The AUROC and TPR is 100% or close to 100% in all settings for both watermarks, indicating a very high degree of separability between the watermarked and non-watermarked images, with otherwise exactly the same generation conditions.

Finally, in Table 3, we report the results for the experiments using an inpainting model for regenerating an image. The most surprising finding is that the Gaussian Shading watermark is still largely recoverable with near-100% detection accuracy even though the generated images are very close copies of unwatermarked real images. Even though Tree-Ring watermarks are harder to recover, and detection accuracy is relatively low (only 50% of generated images are recognized as watermarked), the high AUROC still indicates a high degree of separability.

4.3 Visual Comparison

Figure 3 shows examples of images with semantic watermarks generated with canny edge control. More results for all control methods are shown in Section A in the Supplementary Material. The image examples demonstrate that the image layouts are essentially the same between the clean and watermarked cases. In the case of TRW, where a given z_T is changed to include the watermark, we observe only minor changes to the image. GS, in turn, samples z_T from scratch so that any connection to the clean image's initial latent z_T is lost. Still, the layout of the images is precisely controlled with GS while achieving 100% bit accuracy.

5 Discussion and Conclusion

In this work, we discover a rather surprising finding that may put in question how inversion-based semantic watermarks are really realized in an image. Previously, it has been observed that watermarked images have subtle changes in layout compared to denoised images from a clean, unwatermarked z_T. In this work, however, we find that tight layout control still results in high detectability for both Tree-Ring and Gaussian Shading. This leads us to believe that the role of image layout in realizing semantic watermarks is overestimated. Most surprisingly, however, we find that even when unwatermarked images are *copied* using an inpainting model[6], the watermarks are still largely detectable, albeit with severely degraded p-values and bit accuracies.

[6]Note that also in this scenario, the inpainting denoising starts from watermarked z_T.

ACKNOWLEDGEMENTS

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC 2092 CASA – 390781972 and by the Ministry of Culture and Science of Northrhine-Westphalia as part of the Lamarr Fellow Network.

REFERENCES

Bang An, Mucong Ding, Tahseen Rabbani, Aakriti Agrawal, Yuancheng Xu, Chenghao Deng, Sicheng Zhu, Abdirisak Mohamed, Yuxin Wen, Tom Goldstein, and Furong Huang. WAVES: benchmarking the robustness of image watermarks. In *Proc. of Int. Conference on Machine Learning (ICML)*, 2024.

Hai Ci, Pei Yang, Yiren Song, and Mike Zheng Shou. RingID: Rethinking tree-ring watermarking for enhanced multi-key identification. In *Computer Vision – ECCV 2024*, 2024.

I. J. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker. *Digital watermarking and steganography*. Morgan Kaufmann Publishers, 2002.

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 (ICCV)*, October 2023.

Sam Gunn, Xuandong Zhao, and Dawn Song. An undetectable watermark for generative image models. arXiv:2410.07369, 2024.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 2014.

Yepeng Liu, Yiren Song, Hai Ci, Yu Zhang, Haofan Wang, Mike Zheng Shou, and Yuheng Bu. Image watermarks are removable using controllable regeneration from clean noise. *arXiv preprint arXiv:2410.05470*, 2024.

Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 4296–4304, 2024.

Andreas Müller, Denis Lukovnikov, Jonas Thietke, Asja Fischer, and Erwin Quiring. Black-box forgery attacks on semantic watermarks for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025.

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 (CVPR)*, 2022.

Mehrdad Saberi, Vinu Sankar Sadasivan, Keivan Rezaei, Aounon Kumar, Atoosa Malemir Chegini, Wenxiao Wang, and Soheil Feizi. Robustness of ai-image detectors: Fundamental limits and practical attacks. In *International Conference on Learning Representations (ICLR)*. OpenReview.net, 2024.

Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. Tree-rings watermarks: Invisible fingerprints for diffusion images. In *Advances in Neural Information Proccessing Systems (NeurIPS)*, 2023.

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

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3836–3847, 2023.

Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024a.

Xuandong Zhao, Kexun Zhang, Zihao Su, Saastha Vasan, Ilya Grishchenko, Christopher Kruegel, Giovanni Vigna, Yu-Xiang Wang, and Lei Li. Invisible image watermarks are provably removable using generative AI. In *Advances in Neural Information Proccessing Systems (NeurIPS)*, 2024b.

A VISUAL EXAMPLES

Figure 4 shows examples of images with semantic watermarks generated with layout control.

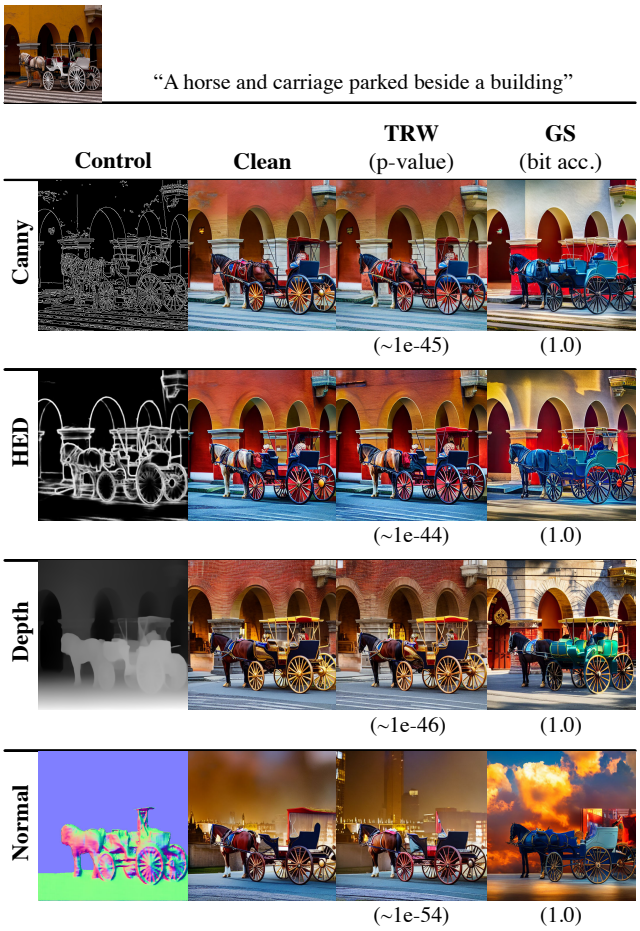

Figure 4: Examples of images with semantic watermarks generated with layout control. The clean images are generated from a randomly drawn z_T. The TRW image are generated from the same z_T with the Tree-Ring pattern inserted.

B ADDITIONAL FIGURES

Figure 5: Comparison between clean and watermarked images. Note that Tree-Ring modifies the existing z_T while Gaussian Shading samples a new one. As a consequence, the rough composition of Tree-Ring watermarked image and clean image is similar, but differs significantly in the arrangement of the details. Gaussian Shading, on the other hand, has no relation to the original image apart from that specified by the prompt.

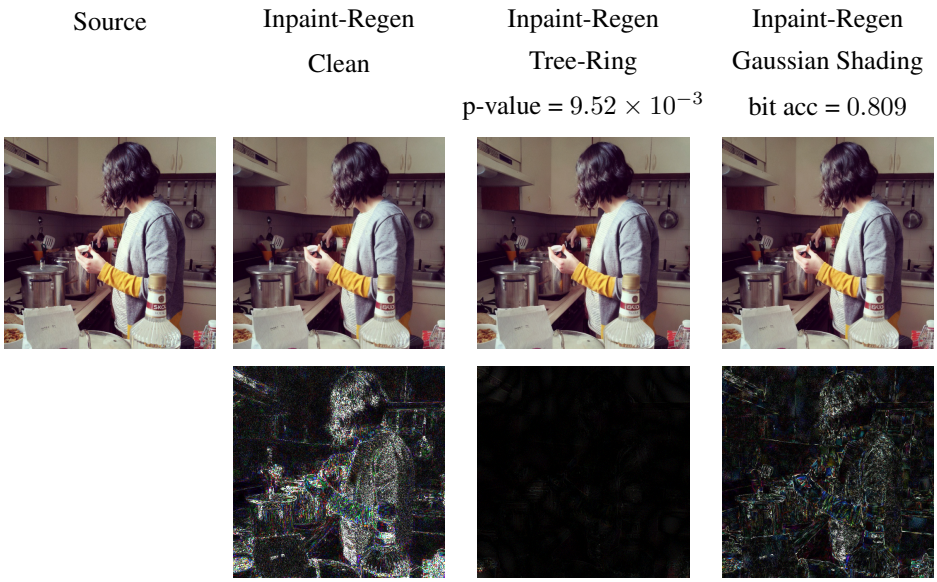

Figure 6: An example of regeneration using inpainting. Left is the real source image from MS-COCO 2017. Second column shows regeneration done by inpainting with an empty mask using SD2's inpainting variant, starting from unwatermarked z_T. Third column shows regeneration but starting from a z_T that has been watermarked using Tree-Ring. Last column shows the result of regeneration starting from a z_T watermarked using Gaussian Shading. P-values and bit accuracies are reported as well, both are above the detection threshold. The second row shows the difference between source and clean inpaint-regenerated (second column), the difference between clean inpaint-regenerated and Tree-Ring-based inpaint-regenerated (third column) and the difference between clean inpaint-regenerated and GS-based inpaint-regenerated (fourth column). All difference are multiplied by 10 for better presentation.

C AUROC AND TPR

Tables 4 and 5 provide further intuition by showing the AUROC and TPR for TRW and GS with different control signals.

	AUROC					TPR@1%FPR				
	None	Canny	HED	Depth	Normal	None	Canny	HED	Depth	Normal
Clean	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
JPG	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
G.N.	1.000	1.000	1.000	1.000	1.000	0.998	0.998	0.998	0.996	0.998
SP.N.	0.999	1.000	1.000	1.000	1.000	0.942	0.976	0.990	0.992	0.984
G.N.+JPG	1.000	1.000	1.000	1.000	1.000	0.998	1.000	0.998	0.998	0.998

Table 4: Tree-Ring Watermarking: AUROC (AUC) and TPR@1%FPR (TPR) for different control types and different perturbation settings.

	AUROC					TPR@10^{-6}FPR				
	None	Canny	HED	Depth	Normal	None	Canny	HED	Depth	Normal
Clean	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
JPG	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
G.N.	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
SP.N.	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
G.N.+JPG	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

Table 5: Gaussian Shading: AUROC and TPR@10^{-6}FPR for different control types and different perturbation settings.