Supplementary Materials

Frequently Asked Questions (FAQs)

✻ Why do we choose to generate 5 paraphrased images using the technique described in [3]?

➠ The choice of generating up to 5 paraphrased images is based on experimentation. Through multiple trials we determined that processing 5 paraphrases achieves a balance between computational efficiency and analytical accuracy. This number was selected as it provides consistent and reliable results while minimizing the computational time required for analyzing each paraphrased image and its salient regions.

Figure 7 illustrates the effect of number of paraphrases on non-melting point generation.

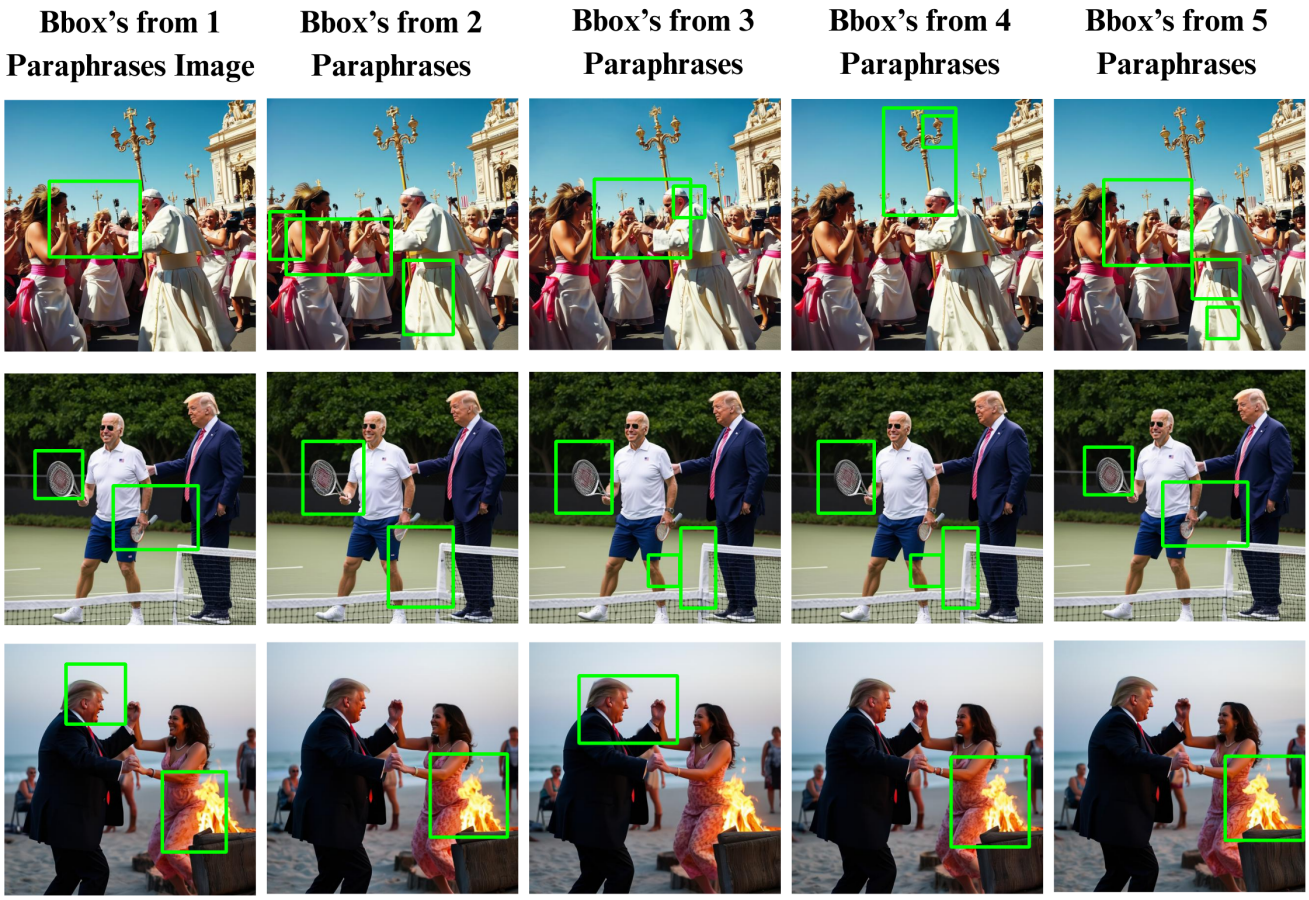

Figure 7. Effect of Number of Paraphrases on Non-Melting Point generation: The plot illustrates the trade-off between the number of paraphrases and the resulting bounding box consistency, showing that increasing paraphrases beyond a certain number yields minimal improvement while significantly increasing processing time.

✻ Why would extrinsic hallucination be riskier?

➠ According to the "extrinsic hallucination" definition, this kind of hallucination does not have any way to verify it from the source prompt. Hence, it is likely to be more harmful than the intrinsic ones.

✻ What is the purpose of constructing Factual Mirage and Silver Lining hallucination data?

➠ We want to show that hallucinations can happen in both cases, factually correct and incorrect prompts. Hence, in this paper, we construct an exhaustive dataset called

✻ Why do you select high-entropy points for mitigation techniques?

➠ High entropy points are more uncertain points in the context of text generation and hence, more likely places where the LLM hallucinates. Hence, our mitigation approach works by detecting and replacing such high entropy points.

✻ Why does comparing and selecting salient regions across different image versions for watermark placement lead to better detection performance?

➠ Analyzing salient regions across multiple image paraphrases allows us to determine which areas of the original image remain stable and consistent despite variations introduced by paraphrasing. Placing the watermark in these robust regions ensures that it is less likely to be distorted or removed by diffusion-based or paraphrasing attacks, thereby improving detection performance.

✻ Why do we compare final results using only 3 different saliency methods?

➠ We produce and compare results using only 3 saliency methods—XRAI, MSI Net, and Vanilla Integrated—due to the high computational cost associated with other methods and resource limitations. These methods were selected for their effectiveness and feasibility within the given constraints.

✻ What is the effect of different saliency thresholds?

➠ Saliency thresholds determine the size and continuity of the regions used for watermarking. A very low saliency threshold (e.g., top 10 or 20) results in the selection of small, discontinuous regions, which are often impractical for watermarking. On the other hand, a very high saliency threshold (e.g., top 80 or 90) would encompass nearly the entire image, undermining the purpose of saliency estimation. The ideal threshold strikes a balance, selecting regions that are continuous but not overly large, allowing for the identification of multiple distinct salient regions for watermarking.

✻ How can the PECCAVI watermarking scheme be integrated with other watermarking methods?

➠ The PECCAVI watermarking scheme is a model-agnostic approach that focuses on identifying and utilizing the non-melting points of an image. This adaptability makes it suitable for integration with other post-processing watermarking techniques.

✻ How do you map the bounding box regions from an image to its latent space representation?

➠ Mapping bounding box (bbox) regions from the image space to the latent space involves scaling the coordinates of the bounding boxes. The bounding boxes in the image space are initially constrained within the bounds of 512×512. These coordinates are then directly scaled down to match the latent space resolution of 64×64.

✻ Why are you embedding watermark in patches rather than the salient bounding boxes itself?

➠ Bounding boxes obtained from the saliency detection method are mostly of various sizes and not a fixed woth and height. Another reason for using patches oof fixed size instead of the bounding boxes is to maintain the spatial domain information by converting that specific patch into fourier domain and then adding watermark to it. This allows us to keep the saptial domain information intact as we are only converting the patch to fourier domain and inverse fourier domain. This inversed fourier domain patch which is now watermarked will go through SDXL model with fixed input size (another reason) to generate the patch again which is then merged with the original image.

✻ Why did we use only $s = 0.1$ and $s = 0.2$ in our paraphrasing experiments?

➠ Empirically, we observed that paraphrased images generated with $s = 0.1$ and $s = 0.2$ retained high visual and semantic similarity to the original, while effectively removing visible watermarks. Increasing the paraphrasing strength s beyond 0.2 introduced significant distortions and yielded images with minimal resemblance to the source, defeating the core purpose of watermark robustness evaluation. Furthermore, higher values of s cause larger shifts in the latent space of the generative model, resulting in semantic drift — which we intentionally avoid to ensure a realistic and fair paraphrasing scenario.

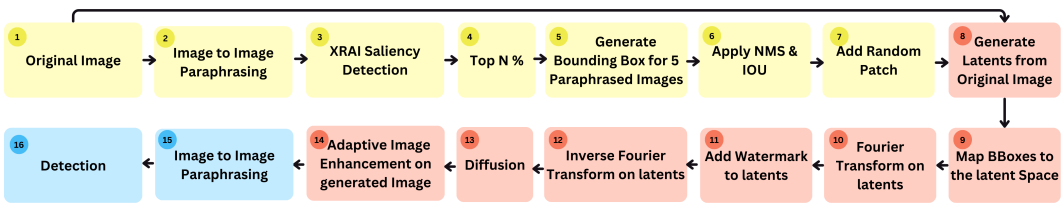

Figure 8. Workflow of the PECCAVI watermarking scheme. The yellow boxes represent the preprocessing steps, red boxes highlight the components that can be replaced or adjusted for integration with other models, and blue boxes indicate the final detection and evaluation steps.

✻ **Why would an attacker use visual paraphrasing when they already have access to a non-watermarked generative model?**

- **Proprietary Content Theft**: Repurposing watermarked content from premium services (e.g., Midjourney, Adobe Firefly) while removing attribution markers.
- **Regulation Circumvention**: Removing compliance markers mandated by regulations like the EU AI Act.
- **Asymmetric Model Access**: The watermarked image may originate from a superior proprietary model (e.g., DALL-E 3) that the attacker cannot access directly.
- Most importantly, our work demonstrates a fundamental vulnerability which is if watermarks can be bypassed through semantic preserving transformations the security guarantee fails regardless of the attacker's specific capabilities.

✻ **Isn't visual paraphrasing just generating a new image rather than removing the watermark?**

- Yes this is analogous to how text paraphrasing circumvents LLM watermarks by generating new text with equivalent meaning. Our contribution demonstrates that watermarking schemes fail when content can be semantically reproduced, which generative models enable. If a watermark cannot survive semantic-preserving regeneration, it cannot fulfill its purpose of persistent content attribution. This represents a fundamental security flaw in the watermarking paradigm.

Appendix

This section provides supplementary material in the form of additional examples, implementation details, etc. to bolster the reader's understanding of the concepts presented in this work.

1. Visual Paraphrase: The Operational Details

Visual Paraphrase is a technique used to create variations in an image that do retain the original sematic content but may have a different visual presentation similar to paraphrasing in natural language processing but with images. The effectiveness of visual paraphrasing can be controlled by adjusting two main parameters:

- **Paraphrasing Strength (s)**: This parameter, ranges from 0 to 1 and determines the extent of deviation from the original image. A value of s close to 1 gives more freedom which allows the new image to be more different from the orginal image and similarly s close to 0 prodcues image which is almost similar to original image given.
- **Guidance Scale (gs)**: This parameter controls how much the new image generated will follow the text caption during the image generation process. A higher guidance scale follows caption given more closely when creating image and vice-versa.

Optimal Parameters: Finding the balance of paraphrasing strength and guidance scale becomes very important and through calculations it is found that low strength values($s \leq 0.4$) and a guidance scale($4 \leq gs \leq 7$) is optimal.

2. Non-Melting Point (NMP) Detection: Operational Details

The identification of Non-Melting Points (NMPs) constitutes a cornerstone of the methodology proposed in this work, representing an innovative approach to watermarking. The authors assert that not all regions within an image are equally significant. To mitigate vulnerabilities to visual paraphrase attacks, as described in [3], the watermark must be embedded across multiple, strategically selected locations termed Non-Melting Points (NMPs).

The procedure begins with the generation of n paraphrased versions of the image to be watermarked (in this study, five paraphrases are utilized). Salient regions for each paraphrased image are then identified using the XRAI algorithm [16]. A filtering step is subsequently applied to retain only the top $k\%$ of salient regions (e.g., the top 30%), effectively discarding irrelevant portions of the image to focus on the most meaningful areas.

Within the identified salient regions of each paraphrased image, bounding boxes are generated. These bounding boxes are aggregated across all paraphrased images to form a comprehensive set of candidate regions. To refine this set, the Intersection Over Union (IOU) metric [11] and Non-Maximal Suppression (NMS) technique [24] are employed. IOU quantifies the degree of overlap between bounding boxes, facilitating the identification of redundant regions, while NMS prioritizes and retains the most salient, non-overlapping regions. This combined procedure ensures that the final set of bounding boxes encompasses only the most significant and non-redundant areas across all paraphrased images.

This refinement step is crucial for two primary reasons:

1. It excludes small bounding boxes, which, if representing less than 1% of the total image area, may compromise watermarking efficacy.
2. It resolves overlapping bounding boxes that could otherwise lead to distortions in Fourier space watermarking, such as unintended pattern interference.

By embedding watermarks within these Non-Melting Points, the proposed approach enhances robustness against adversarial attacks while preserving the structural integrity of the image.

3. Watermark Strength: The Operational Details

This paper introduces a novel concept of varying the strength of watermarks based on the significance of different regions within an image. As mentioned earlier, not all sections of an image are non-melting points, and consequently, not all watermarks need to be of equal strength. A watermark embedded in a highly prominent region will be detected by most saliency detectors and, therefore, is more likely to face frequent dewatermarking attacks. In contrast, watermarks in less prominent areas are less susceptible to such attacks.

To address this, multiple paraphrased versions of the original image are processed through a saliency detector, generating bounding boxes around the detected salient regions for each version. A function is applied to merge bounding boxes in nearby areas and remove any redundancies. Additionally, a Non-Maximal Suppression (NMS) algorithm is used to eliminate overlapping boxes, ensuring only the most relevant regions remain.

Once the final set of bounding boxes is established across all paraphrased images, a scoring function evaluates how consistently each region is identified as salient. For example, if a particular region is highlighted as salient in all five

paraphrased versions, it is assigned a score of 0.1. Conversely, if the same region appears salient in only two images, it receives a higher score of 0.75. Based on these scores, different watermark strengths are applied to each bounding box, ensuring that regions with higher exposure receive more robust watermarking, while less prominent regions are marked more subtly.

3.1. Fourier Space Watermarking

Rather than directly embedding the watermark in the pixel or latent space, the proposed architecture applies a Fast Fourier Transform on the latent space before mapping the bounding boxes to the latent space from the normal image and embedding the watermark in them. These watermarked latents are passed through the Stable Diffusion Model to generate watermarked images. This method provides two key advantages:

1. It allows us to condition the latent vector in such a way that the image quality remains undisturbed, and no visible patterns appear in the final image.
2. Embedding the watermark in the Fourier domain inherently enables resistance to common dewatermarking attacks such as brightness changes, scaling, and compression. This is because the Fourier transform exhibits invariance to these types of transformations, as demonstrated in the following equations:

For brightness changes:

$$F(I(x, y) + b) = F(I(x, y)) + \delta(f = 0)$$

where F is the Fourier transform, $I(x, y)$ is the image, and b is the brightness offset. The delta function $\delta(f = 0)$ indicates that brightness changes only affect the zero-frequency component, leaving the rest of the Fourier domain invariant.

4. Random Patching: Operational Details

To make watermarking more robust against potential attackers who may attempt to explicitly detect the salient regions of an image, an additional layer of unpredictability is introduced through the use of random patching. A random patch is added to the image prior to embedding the watermark. This patch is designed to match the size of the largest Non-Melting Point (NMP) in the image and is strategically positioned to avoid overlapping any existing NMPs.

The random patch operates identically to the other NMPs, but its non-deterministic nature adds a layer of unpredictability to the watermarking process. This makes it significantly more challenging for attackers to isolate and remove watermarked regions, as they cannot reliably identify all salient regions based on deterministic patterns.

5. Noisy Burnishing: Operational Details

Another potential attack vector involves explicitly detecting salient regions within the image and focusing watermark-disrupting attacks on these regions. To mitigate such attempts, adversarial noise is added to the watermarked image. This process, referred to as noisy burnishing, disrupts the detection of salient regions in the image, as proposed in [10].

Since the watermarks are embedded in the Fourier domain, the addition of noise in the spatial domain does not interfere with the watermark itself. As a result, noisy burnishing effectively reduces the saliency detection accuracy of attackers while preserving the integrity of the watermark, ensuring high detection rates.

6. PECCAVI: Watermark Detection

The PECCAVI watermarking scheme demonstrates resilience against traditional de-watermarking methods and maintains notable robustness against paraphrasing attacks compared to other state-of-the-art watermarking approaches. This robustness is achieved by embedding watermarks in regions that remain consistent across image paraphrasing processes, ensuring the watermark's integrity is preserved despite paraphrasing transformations. When subjected to high-strength paraphrasing attacks (e.g., s ≥ 0.5), the resulting image differs substantially from the original, thus failing to convey the original image's semantic meaning. In such cases, the preservation of the watermark becomes less relevant, as the semantic divergence between the original and modified images renders them effectively distinct.

7. Distortion vs. Detectability

PECCAVI shows a higher level of distortion in comparison to Zodiac, though not visible to the naked eye, but this increase in distortion is accompanied by a tradeoff in detectability. Across all components measured, PECCAVI consistently outperforms Zodiac by at least 10% in terms of detectability. This improvement makes PECCAVI suitable for applications where detectability is prioritized over minimizing distortion. Such tradeoffs are an inherent aspect of system optimization, where enhancing one parameter often requires diminishing another.

Figure 9. The above images showcase a comparison of Original Images vs PECCAVI Watermarked Images.

8. Standardized Detection Protocol

For multi-bit watermarking methods (Gaussian Shading, Stable Signature, DwtDctSVD), WDP corresponds to the Bit Accuracy of the recovered payload. For zero-bit or hypothesis-testing methods (PECCAVI, ZoDiac, Tree-Ring), WDP corresponds to the detector confidence score $(1 - p)$.

9. Results

We compare the performance of various various image watermarking methods ([38], [30], [9], [23], [22]) , [6] under attacks such as brightness enhancement (factor 0.5), Gaussian noise (std 0.05), JPEG compression (quality 50) and visual paraphrasing ([3]) using stable-diffusion-xl-base-1.0 (paraphrase strengths 0.2, 0.4, 0.6, 0.8 and 1.0). The table highlights the vulnerability of current image watermarking systems to the visual paraphrase attack.

9.1. Detailed Attack Configurations

To ensure standardized comparisons, all baseline watermarking methods were evaluated using their official implementations with default parameters and payload capacities. For the VAE-based generative attacks (BMSHJ18, CHENG20, ZHAO23), target images were resized to a fixed resolution of 512×512. Compression was applied using Mean Squared Error (MSE) optimization at a default quality level of $q = 3$ (selected from a standard range of $q \in \{1..6\}$). For the diffusion-based generative attacks, we employed the `stable-diffusion-2-1` pipeline (fp16 revision) configured with a noise step of 60 and a batch size of 5. Traditional attacks were applied with the following parameters: JPEG compression at quality 50, brightness and contrast adjusted by a factor of 0.5, Gaussian noise added with a standard deviation of 0.05, Gaussian blur applied with a kernel size of 5 and $\sigma = 1$, and rotational attacks executed at $90°$.

We report results for most effective PECCAVI configurations for the diffusion db dataset [32]. The experiments were conducted on 100 images from the DiffusionDB dataset, with the saliency threshold (λ) set to the top 30. Table 4 showcases the corresponding results.

10. Computational Comparison

In this paper, we evaluate three state-of-the-art watermarking techniques: PECCAVI, ZoDiac [38], and Watermark Anything (WAM) [29]. PECCAVI and ZoDiac are latent-space-based watermarking methods applied post-image generation by AI models, while WAM, developed recently by Meta FAIR, embeds watermarks directly in the spatial domain.

The computational FLOPs required for these techniques vary significantly. PECCAVI requires approximately 1.872×10^{17} FLOPs per image due to certain saliency detection components that run on the CPU. ZoDiac requires around 1.4976×10^{17} FLOPs to produce a final watermarked image. In contrast, WAM is significantly faster, requiring only about 1.872×10^{15} FLOPs to complete the watermarking process. However, performance under paraphrase attacks reveals that the computational overhead of PECCAVI and ZoDiac is justified by their superior detection accuracies: PECCAVI achieves 99.8%, ZoDiac achieves 89%, and WAM achieves only 45%.

Figure 10 visually illustrates the computational FLOPs and detection accuracy of these techniques, highlighting the trade-offs between speed and robustness in watermarking methods.

Table 4. PECCAVI Results on DiffusionDB Dataset

Method	Image Quality			Avg. Watermark Detection Probability (WDP)				Paraphrase ($s = 0.1$)	Paraphrase ($s = 0.2$)
	PSNR	SSIM	FID	Brightness	Gaussian Noise	Rotation	JPEG		
PECCAVI (MSI Net)	29.66	0.93	26.41	0.98	0.99	0.85	0.92	0.83	0.82
PECCAVI (XRAI)	29.49	0.93	30.17	0.98	0.98	0.87	0.92	0.89	0.86

Watermarking Method	DwtDctSVD	HiDDen	Stable Signature	Tree Ring	ZoDiac	Gaussian Shading	RingID	VINE
Pre-Attack	0.98	1.00	0.99	1.00	1.00	1.00	1.00	0.99
Brightness	0.01	0.75	0.75	0.98	0.92	0.99	0.99	0.97
Rotation	0.96	0.93	0.98	0.92	0.91	0.93	0.97	0.98
JPEG Compression	0.65	0.88	0.65	0.97	0.89	0.94	0.98	0.95
Gaussian Noise	0.14	0.91	0.73	0.98	0.90	0.93	0.94	0.98
Visual Paraphrase (Ours)								
$s = 0.2$	0.00	0.298	0.51	0.683	0.70	0.711	0.77	0.128
$s = 0.4$	0.00	0.215	0.225	**0.394 (73% ↓)**	0.335	**0.384 (85% ↓)**	0.443	0.092
$s = 0.6$	0.00	0.154	0.176	**0.255 (35% ↓)**	0.219	**0.221 (42% ↓)**	0.361	–
$s = 0.8$	0.00	0.096	0.107	**0.156 (39% ↓)**	0.140	**0.157 (28% ↓)**	0.234	–
$s = 1.0$	0.00	0.041	0.059	**0.097 (38% ↓)**	0.065	**0.119 (24% ↓)**	0.107	–

Table 5. Watermark detection rates (η) for various methods on the COCO dataset [21], pre-attack and post-attack. Detection falls sharply with increasing s. Tree Ring and Gaussian Shading show the most resilience.

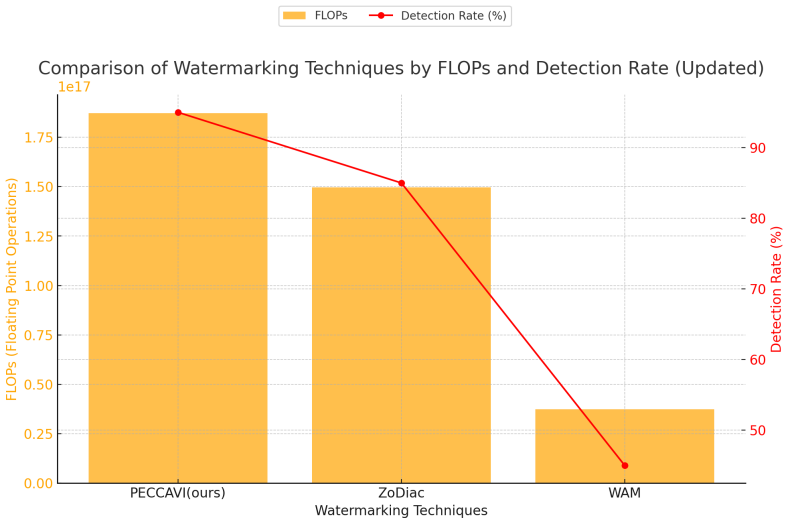

Figure 10. Comparison of Computational FLOPs and Detection Accuracy for Watermarking Techniques. PECCAVI and ZoDiac operate in the latent space, whereas WAM works in the spatial domain.

11. Discussion & Limitations of PECCAVI

We identify two key limitations: (i) computational overhead, and (ii) challenges associated with open-ended paraphrasing.

The proposed method operates as a post-processing technique, whereby the watermark is embedded into the image after generation. This approach necessitates the preparation of image paraphrases and the identification of salient regions across these paraphrases, which significantly increases computational requirements. Future work could focus on optimizing computational efficiency and transitioning the approach to a generative paradigm. Furthermore, since the method involves embedding multiple watermarks in the Fourier domain of the image, careful consideration must be given to ensure that watermark patterns do not overlap across channels, as such overlap can result in noticeable image distortions.

11.1. Compositional Forgery

Since PECCAVI operates as a zero-bit provenance watermark, transplanting a watermarked NMP onto a malicious background will still cause the detector to flag the image as AI-generated. PECCAVI is not designed for localized ownership verification but remains robust for provenance detection under compositional manipulation.

12. Examples on visual paraphrase strength variation

The figure 11 showcases examples of visual paraphrasing at different strength levels. The images illustrate how varying the strength parameter impacts the degree of transformation applied to the original image. Lower strength values result in paraphrases that closely resemble the original, while higher strength values introduce more significant alterations. Figure 11.

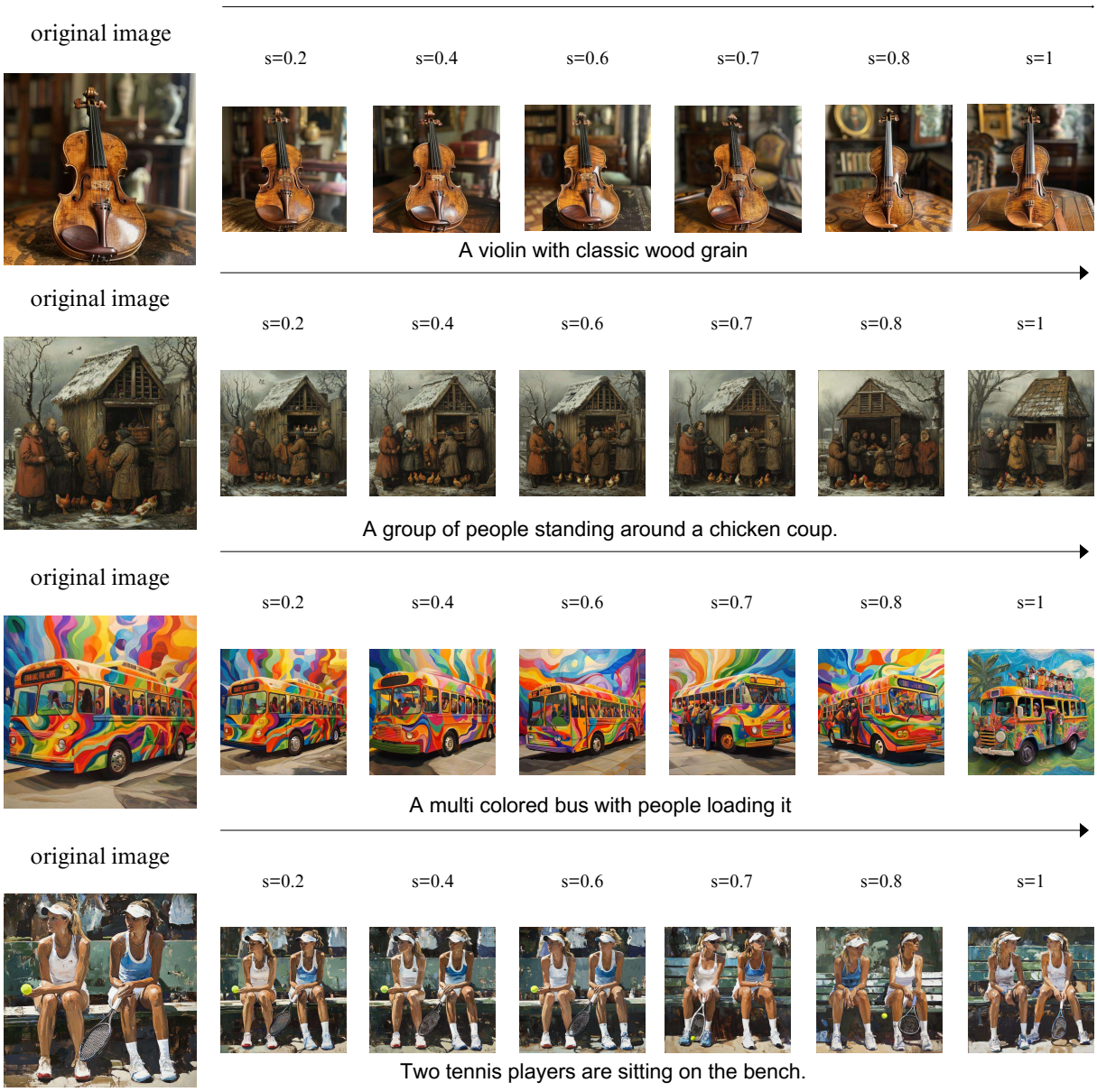

Figure 11. Examples of Visual Paraphrasing with varying levels of strengths.

13. Impact of Strength and Guidance Scale on Watermark Detectability and Quality

Figure 12 illustrates the relationship between the CLIP Maximum Mean Discrepancy (CMMD) score and the detectability of visual paraphrases as influenced by variations in strength and guidance scale.

13.1. Semantic Fidelity Verification

To ensure that VPA preserves semantic content rather than generating unrelated images, we evaluate CLIP-based similarity (CMMD) and conduct a Mean Opinion Score (MOS) study.

For paraphrasing strengths $s \leq 0.2$, semantic similarity remains high (SSIM > 0.85) while watermark detectability drops significantly. These results confirm that VPA operates in a semantic-preserving regime rather than arbitrary image regeneration.

14. Visual Paraphrase Acceptability in MOS Evaluation

Figure 13 presents a set of visual examples illustrating both accepted and rejected paraphrases during the MOS (Mean Opinion Score) evaluation. These examples highlight the differences in image quality and semantic consistency that led to their respective ratings. Accepted paraphrases maintain a high degree of similarity to the original image while preserving key visual and contextual elements. In contrast, rejected paraphrases exhibit significant deviations that detract from the original image's meaning or visual quality, resulting in lower MOS ratings. This comparison underscores the criteria used by human evaluators to assess the acceptability of visual paraphrases.

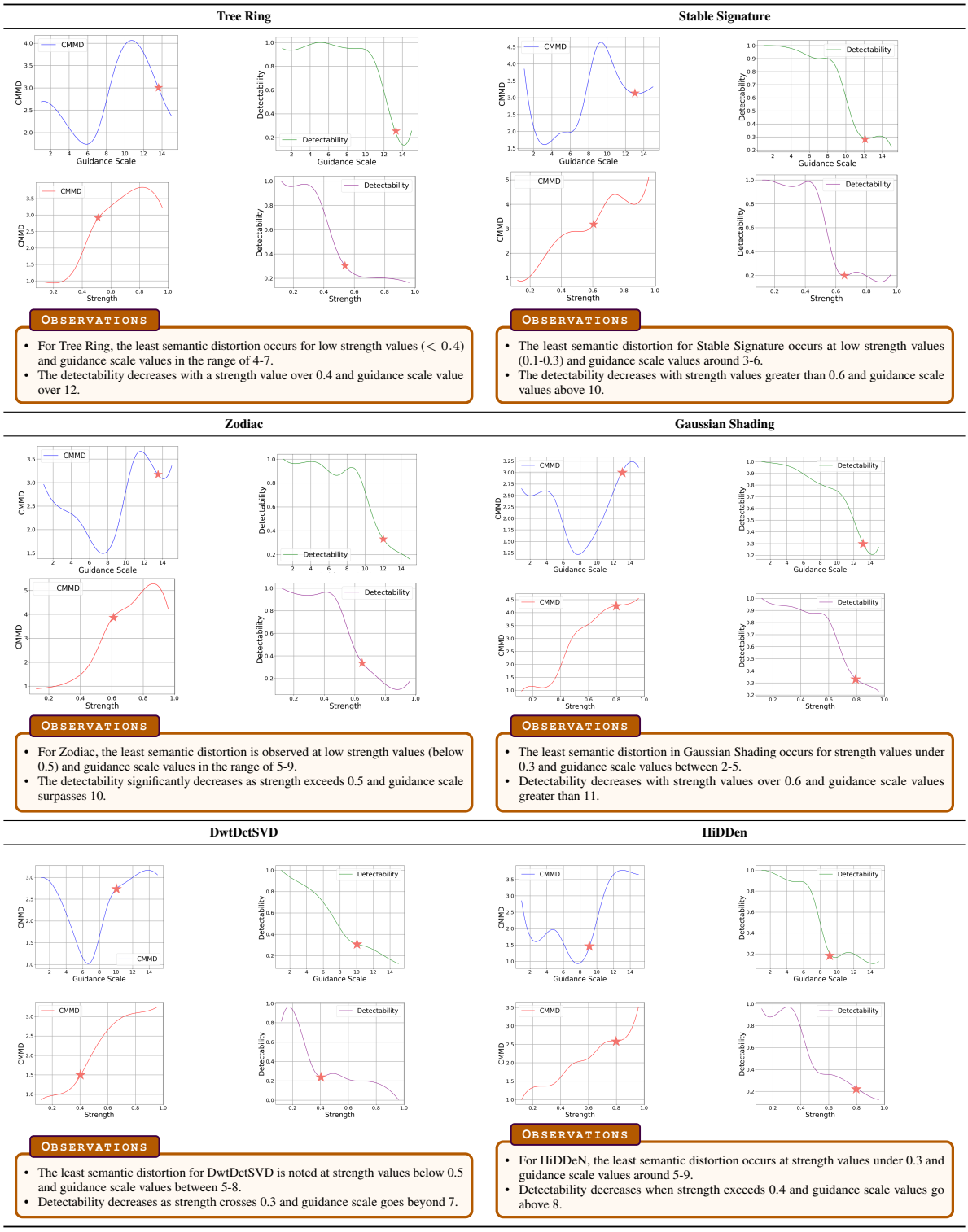

Figure 12. This figure shows the variation of CMMD [15] and detectability of visual paraphrases with respect to strength (s) and guidance scale (gs). The symbol indicating the optimal s and gs value for the particular technique is shown in the figure legend.

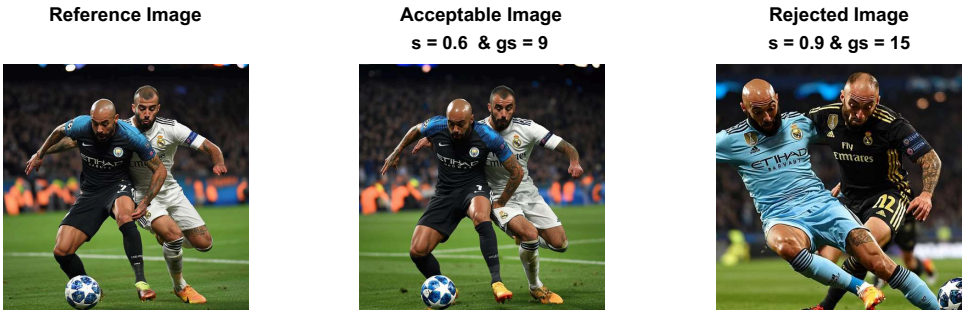

Figure 13. Examples of acceptable and rejected Visual Paraphrasing during MOS evaluation.