

Detailed Response to the ICCV Reviewers

Reviewer rQjt

Simple approach but a similar idea has been explored

Paper Summary:

This paper proposes a method to safeguard copyright content from text-to-image model generations. The method consists of three main steps: (1) use text feature similarity and query LLM to detect whether the user text prompt contains copyrighted materials, (2) query LLM to update the prompt iteratively until all copyrighted concepts are removed, and (3) blend the updated and the original prompt embedding to generate content. The advantage of this approach is that it is an inference-time algorithm without model weight update, so the modified content maintains high fidelity, which is shown in the CLIP similarity metric and qualitative figures. Ablations are done to validate the design choices.

Paper Strengths:

1. The idea is intuitive and well-motivated. The inference-time approach allows modified content to maintain high fidelity. Moreover, because this approach does not modify the model itself, intuitively the method can support a larger set of copyrighted concept to shield without sacrificing the model's performance.
2. The writing is clear and easy to follow, and the reasoning behind each design choice is well-motivated.

Major Weaknesses:

1. Applying inference-time correction to protect copyright content has been explored by Safe Latent Diffusion [1]. It would be great to compare this method.
2. Many related works in concept removal are not cited and compared [2, 3, 4]. It is difficult to fully assess the effectiveness of the proposed method without comparisons to potential baselines.
3. The prompt correction mechanism is completely text-based, and a potential way to jailbreak this is to curate unrelated prompts that can still generate sensitive/copyrighted content [5].

[1] Schramowski et al. Safe Latent Diffusion: Mitigating Inappropriate Degeneration in Diffusion Models. [2] Gandikota et al. Unified Concept Editing in Diffusion Models. [3] Kumari et al. Ablating Concepts in Text-to-Image Diffusion Models. [4] Zhang et al. Forget-Me-Not: Learning to Forget in Text-to-Image Diffusion Models. [5] Yang et al. SneakyPrompt: Jailbreaking Text-to-image Generative Models.

Minor Weaknesses:

1. Given the proposed method, the scope of this project does not necessarily need to be restricted to copyright shielding. Could it be applied to cases such as violence or NSFW content?
2. In some figures, we can still see traces of copyright content after applying the method. For example, in Figure 1 (pikachu), the 3rd image in the bottom row still arguably looks similar to a pikachu.

Answer 1: We have compared GoG with SLD in Fig. 1 and GoG is superior to SLD in both image quality and copyright shielding ability. SLD being originally designed for DDPM based architectures, does not perform well when applied to rectified flow matching MM-DiT models like Flux.1-dev, as evident from our testing.

Answer 2: We have added all these citations in the revised paper.

We compare the efficacy of our method against two baselines ESD \cite{gandikota2023erasing}, and UCE \cite{gandikota2024unified}, across UNet and transformer backbones. To ensure fair evaluation on the Flux transformer backend, we use the official implementations of ESD and UCE. As ESD supports only UNet models, it is evaluated on SD and SDXL; UCE supports both and is evaluated on all. Figure~\ref{fig:exp_grid} shows qualitative generations across four target concepts: Batman, BMW, Chris Hemsworth, and Starry Night. ESD exhibits degradation and fails to preserve fidelity. UCE often leaks concept traces and underperforms. In contrast, GoG achieves consistent erasure with minimal leakage, particularly in natural image and style transfer tasks. Table~\ref{tab:method_metric_comparison} supports this: GoG yields the highest CLIP-I similarity (preserved generality) and higher LPIPS (realism). Notably, $\mathrm{GoG}_{\{\text{flux}\}}$ strikes the best balance, outperforming both baselines.

Answer 3: The threat model assumed in our paper is that of inadvertent violation of Copyright material. Jailbreaks is a different threat model and mitigation strategies against jailbreaks can be added to the proposed GoG work. Furthermore, it should be noted that SneakyPrompt seeks adversarial prompts (p_a) using RL algorithms to bypass filters ($\mathcal{F}(\mathcal{M}, p_a) = 0$) while preserving target semantics in $\mathcal{M}(p_a)$. GoG's architecture (detection f_{detect} , iterative rewriting f_{rewrite} , and adaptive guidance $f_{\text{adaptive_CFG}}$) complicates the reward landscape of such form of attack, and its anticipated the number of queries that will be required to be made to the model to learn adversarial prompt modification will be significantly higher. Moreover, as the number of queries increase, it becomes increasingly feasible to detect a malicious or anomalous source of queries.

Answer M1: GoG's detector could flag any policy list, but the rest of the pipeline is tuned for copyright. We **rewrite** prompts to preserve user intent while stripping specific IP cues, a strategy that makes little sense for violence or NSFW, where the correct action is usually outright refusal or a dressed person. Our similarity band and legal audit are likewise calibrated to copyright law, not safety guidelines. Adapting GoG to those domains would require a different objective and new thresholds, so we deliberately restrict the scope to copyright shielding here.

Answer M2: The 3rd image is in fact more closely resembling Pikachu, the ideal and foolproof range for GoG kicks in at $\alpha=0.7$ onwards, as mentioned in the paper. Some of the images, such as the one pointed out, are generated with ranges between 0.5-0.65, which may cause the output to more closely resemble the original character.

Reviewer niEm22

review of GoG

Paper Summary:

This paper introduces Guardians of Generation (GoG), a model-agnostic framework for inference-time copyright protection in text-to-image generative models (e.g., Stable Diffusion, SDXL, Flux). The key innovation is a three-part system: (1) a protected concept detector that

flags prompts likely to elicit copyrighted content, (2) an LLM-based prompt rewriter to sanitize the prompt while preserving user intent, and (3) an adaptive Classifier-Free Guidance (CFG) mechanism that blends the original and rewritten prompts during sampling. Unlike prior methods that require retraining or model modifications, GoG operates entirely at inference time, providing a practical plug-and-play solution. Extensive evaluations show that GoG reduces copyright-infringing outputs while maintaining semantic fidelity and visual quality.

Paper Strengths:

Timely and Relevant Problem This addresses a significant concern regarding copyright infringement in generative AI, which faces increasing public and legal scrutiny.

Model-Agnostic and Inference-Time Being prompt-based, the system operates without retraining or internal modifications, making it widely applicable and easy to integrate.

Robust Detection Pipeline It effectively combines embedding-based similarity detection with large language model (LLM) policy judgment to identify both explicit and subtle prompt violations.

Creative Adaptive Guidance Interpolating original and sanitised prompt embeddings allows for precise control over the balance between semantic fidelity and compliance.

Comprehensive Experiments The system is validated across three popular models using nuanced metrics (CLIP, LPIPS, SSIM, CONS, DETECT) and various prompts, including indirect anchoring and complex scenarios.

Major Weaknesses:

Computational Overhead Latency significantly increases, with generation times five to seven times longer. This delay can become a bottleneck for real-time applications or large-scale deployments.

Dependence on LLM Quality The success of the rewriting process heavily relies on the quality of the underlying language model (LLM) and the effectiveness of the prompt engineering. Performance may vary between different versions of LLMs.

Fixed Prompt List for Detection The detection mechanism is based on a predefined and limited set of protected concepts. This approach may struggle to adapt to rapidly changing cultural or legal definitions of what constitutes protected content. How would this list be distributed?

Limited Exploration of False Positives/Negatives There is a lack of analysis regarding incorrectly flagged prompts and the effects of over-sanitisation on user intent and image diversity.

Minor Weaknesses:

Some formatting and clarity issues (e.g., formula layout on page 4, slight repetition in narrative).

A more user-centric qualitative evaluation (e.g., human preference studies or user interviews) would enhance the real-world relevance.

The method's behaviour under adversarial prompting (e.g., evasion tactics) is not deeply explored beyond "indirect anchoring".

Answer 1,2,3: The additional time cost comes from **three plugins**: LLM based detection, LLM rewrite, and one extra mixed-prompt pass in adaptive CFG. Batched prompts, cached embeddings, and speculative decoding can cut this time cost 2x-3x (although we have not explored that in this paper). However, the payoff is permanent: **zero retraining, instant policy edits, and no fidelity loss** on clean prompts, benefits concept unlearning/editing methods can't match. GoG's LLM dependence is an asset and not liability as each new, faster, cheaper model instantly upgrades GoG's rewrite quality without retraining. Better LLMs just need fewer iterations

(Section 3.2). The concept list is a hot-swappable JSON feed that easily adapts to new IP or regional laws without downtime. Hence the modest, evenly distributed runtime premium is a practical trade-off for a maintenance-free, policy-flexible shield at scale.

Answer 4: We have done this analysis and shared the results in Fig.8 in the revised paper.

Reviewer FnY722

Interesting work, but lack of comparison to previous work

Paper Summary:

The paper presents a method for guiding generations away from protected concepts. The method detects if a prompt is within a set of protected concepts, rewrites it with an LLM, and generates an image with the overwritten prompt. The method has parameters to tune the sensitivity on the protected concepts.

Paper Strengths:

- The paper is tackling an interesting, open problem in responsible AI.
- The method is clearly described
- The method is "plug and play", and can be a preprocessing module in front of different backend generators.
- The paper thoroughly ablates/sweeps the its parameters in the results section.

Major Weaknesses:

Q1: A major issue in the paper is the lack of comparison to previous work. The paper is not creating a new problem. In fact, the paper does a thorough job listing references for different types, including both unlearning and prompt-based methods (albeit with some methods missing). However, it does not compare to any of them in the results section.

Q2: While the paper evaluates the method across different metrics in Tables 2-5, it is difficult to understand the meaning of the methods without any comparison. As such, the paper claims the "metrics should ideally lie in a moderate, balanced range" (L360), without a frame of reference of what that means.

Q3: Table 1 compares the nature of different protection strategies, concluding that the proposed method has the "best of all worlds", as indicated by the checkmarks. However, an issue with prompt rewriting strategies is that it does not protect against copyright concept generation if the model weights are released. This limitation should at least be mentioned in the table and in the paper, as the paper is presenting this family of methods as if they are the complete solution to the issue.

Q4: While the paper evaluates how well prompts can be rewritten/generated when they hit a protected concept, I did not see an evaluation on if there were "false positives". That is, do innocent prompts get accidentally caught and rewritten, resulting in generations that do not follow the user intent?

Minor Weaknesses:

I believe reference [14] is not listed correctly and should be Rohit Gandikota, Joanna Materzynska, Jaden Fiotto-Kaufman, David Bau. Erasing Concepts from Diffusion Models. ICCV 2023.

[a] Kumari et al. Ablating Concepts in Text-to-Image Diffusion Models. ICCV 2023.
<https://arxiv.org/abs/2303.13516> [b] Zhang et al. Forget-Me-Not: Learning to Forget in Text-to-Image Diffusion Models. CVPR 2024.

Q5: The timing in Table 8 is appreciated. It would be helpful to understand where the extra time comes from, broken down by the different steps that are added. In addition, how much time is added for a prompt that does not violate any of the protected concepts?

Minor typos

- Algorithm 1 L5 is missing a)
- Figure 6 should say "w/ GoG" instead of "CoG"

Author Response

Answer 1: Thank you for bringing this up, we have now inserted a side-by-side visual benchmark (see Fig R1) comparing GoG to three baselines: Safe Latent Diffusion (prompt filtering), Erase-a-Concept (weight editing), and a full unlearning finetune, all on identical SD 2.1 prompts. The images show that GoG removes protected cues while keeping stylistic intent, whereas SLD leaves residual IP elements and Erase/Unlearn sacrifice prompt fidelity. Because these methods optimise different objectives, a single metric table could mislead; nonetheless we report SSIM on the shared set (GoG = 0.36 vs. SLD \approx 0.22 with CLIP-I(GoG) = 0.85 and CLIP-I(SLD) = 0.76) to illustrate the trade-off. Full code and prompts are provided in the appendix so the results can be reproduced or extended.

Answer 2: Copyright shielding must strike a middle ground: **too similar** \rightarrow **infringement; too different** \rightarrow **lost utility (concept unlearning)**. Because standard metrics lack that context, we calibrated them empirically. We did an audit with 5 IP lawyers/paralegals (~50 images each, three conditions: unshielded, GoG, over-sanitised), their "infringing vs. compliant" labels formed two metric clusters. The "compliant yet useful" cluster (e.g., CLIP-T \approx 0.10–0.20, LPIPS \approx 0.15–0.35, DETECT \leq 5) is what we call the **balanced range**. As per the IP lawyers, these cut-offs are heuristic, human review remains essential, and can shift with model or concept, so we report scores relative to this band rather than fix hard numbers. We have provided full evaluation code and prompts in the supplementary material; you are welcome to reproduce these results and verify the thresholds.

Answer 3: We agree with you and will add this point in the camera ready version: **Add a new row in Table 1:** "Robust after weight release?" \rightarrow GoG \times , unlearning \checkmark , retraining \checkmark . **Add a line in Limitation:** "GoG is effective for hosted APIs where weights stay private; if weights are released, weight-level defences (e.g., concept unlearning) remain necessary."

Answer 4: For FN and FP analysis, we constructed two 100-prompt auxiliary datasets (10 concepts \times 10 gradations). *FN data suite:* Each concept has prompts scored 1 \rightarrow 10, where 1 is an obscure description with similarities to the concept, and 10 is maximally direct. every prompt hints at the protected concept with or without naming it, probing missed violations. *FP data suite:* Prompt scored 1 \rightarrow 10, with 1 being a slightly relevant reference, and 10 being an unrelated concept. We use the ChatGPT 4o-mini model as evaluator. We observe a 5% error rate in the initial flagging. Obscure references to certain concepts have a greater probability of evading the LLM filter. Nevertheless, these uncommon references are generally improbable to trigger copyrighted content generation. The 15% FP rate aligns with the anticipated range for superficial mentions of specific concepts. Figure 2 shows the FP and FN rate per concept and as per level of prompt directness.

Answer 5: We report the breakdown of time cost in Table 2 in the form $[Total = Concept\ detection + Prompt\ Rewriting + Adaptive\ CFG + SD/Flux/SDXL]$. SD 2.1 $[185.27 = 57.448 + 16.284 + 75.698 + 35.84]$, SDXL $[187.77 = 57.448 + 16.284 + 75.118 + 38.92]$, FLUX.1-dev $[251.02 = 57.448 + 16.284 + 120.658 + 56.63]$. For *non-protected concepts*, the time costs are of the form $[Total = Concept\ detection + SD/Flux/SDXL]$. SD 2.1 $[93.288 = 57.448 + 35.84]$, SDXL $[95.448 = 57.448 + 38.92]$, FLUX.1-dev $[114.078 = 57.448 + 56.63]$. We will add these in the final version.

Feature	GoG	SLD	UCE	CA	FMN
Inference-Time (No Weight Change)	✓	✓	✗	✗	✗
Dynamic Prompt Handling	✓	✗	✗	✗	✗
Specific Entity Protection Focus	✓	✗	✓	✓	✓
General Unsafe Concept Focus	✗	✓	✓	✗	✓
Concept Erasure/Ablation/Forgetting	✓ (via Rewriting)	Partial (Suppression)	✓	✓	✓
Simultaneous Multi-Concept Editing/Handling	✓ (Prompt-level)	Limited (Single "unsafe" vector)	✓	✓	✓
Requires Anchor/Target Concept for Edit/Guidance	✓ (Protected concepts)	✓ (Unsafe concept)	✓ (Edit/Preserve/Target concepts)	✓ (Target/Anchor or concepts)	✓ (Forgetting concepts)
Uses Fine-Tuning (for editing/guidance)	✗	✗	✗	✓	✓
Aims to Preserve User Intent When Modifying	✓	✗	Partial (via Preservation)	Limited (Focus on Anchor)	Limited (Focus on Forgetting)
Model Agnostic	✓	✗	✗	✗	✗

Table 1. Shows an empirical comparison of different methods and their advantages/disadvantages over Guardians of Generation. SLD: Safe Latent Diffusion, UCE: Unified Concept Editing, CA: Concept Ablation. FMN: Forget-Me-Not

Model	Time w/o GoG	Time with GoG	Total GoG	Detecti on Time	Rewriting Time	CFG Time
SD 2.1 [33]	35.84	185.27	149.43	57.448	16.284	75.698
SDXL [29]	38.92	187.77	148.85	57.448	16.284	75.118
FLUX.1-dev [12]	56.63	251.02	194.39	57.448	16.284	120.658

Table 2: Shows an empirically calculated time-cost breakdown for GoG.