Seeing Is No Longer Believing: Benchmarking Synthetic Image Detection

Succinct Labs

***Abstract*— As generative AI produces increasingly photorealistic images, distinguishing synthetic content from authentic photographs has become a pressing technical and societal challenge. Yet the performance of current detection methods against modern generators, and their robustness to adversarial attack, remains poorly understood. We present AdversIm, a large-scale benchmark dataset spanning multiple generators, image domains, and editing conditions. Using this dataset, we conduct a systematic evaluation of leading detection methods, finding significant variation in accuracy across generators and domains. We further demonstrate that simple adversarial perturbations can substantially degrade detector performance, even for state-of-the-art models. Our results suggest that existing detection approaches face fundamental limitations, and that reliable trust infrastructure will require new technical paradigms.**

I. Introduction

Rapid advances in generative AI have made it possible to mass-produce synthetic images indistinguishable from reality without specialized expertise or equipment, leading to the proliferation of AI-generated content (AIGC). This capability poses an asymmetric threat to institutions that depend on visual authenticity, from newsrooms, to courts, to media platforms. As the speed and scale of synthetic media generation continues to increase, the development of robust detection mechanisms has become an urgent technical and societal imperative.

These concerns are not theoretical. Recent reporting has documented a rise in receipt fraud following the release of advanced image generation models, as employees submit AI-generated expense claims for reimbursement [8]. In the political sphere, campaigns have begun deploying AI-generated deepfakes of opposing candidates [17]. Gig economy platforms have seen workers use synthetic images to fraudulently verify task completion [16]. As generative AI capabilities improve and become more accessible, we expect such cases to proliferate across domains where visual evidence has traditionally served as a trust mechanism.

A leading proposed countermeasure involves training neural network classifiers (referred to as AIGC detectors) to distinguish AI-generated content from authentic photographs (e.g., TruthScan [38]). These classifiers are now commercially deployed by platforms and enterprises, promising automated detection at scale. Vendors often claim high accuracy rates, and a growing ecosystem of APIs and tools has emerged to meet demand from content moderators, insurers, and trust and safety teams. Yet the real-world efficacy of these systems remains unclear, particularly as generators improve.

Further, neural network classifiers are known to exhibit brittleness under adversarial perturbations [3, 27, 36]: modifications imperceptible to human observers can induce misclassification of images previously classified correctly. While incorporating adversarial examples into training data can partially mitigate this vulnerability [36], this approach establishes an ongoing arms race wherein classifiers must continuously expand their training distributions as novel generative models and attack vectors emerge.

The other leading proposed countermeasure is to embed imperceptible "watermarks" in AIGC that serve to identify content generated by a specific model (Google's SynthID [15] is one instantiation of this approach). AI watermarking can be a more effective tool for detecting images generated by *specific* models than generic AIGC detectors, since the latter address the more general problem of identifying images coming from *any* model. However, not only is AI watermarking also susceptible to adversarial attacks [21], but it presents a coordination problem for AI labs: watermarking is of limited use for the AIGC problem if only one or a few of the leading AI models produce watermarks, since a malicious actor can freely choose which model it uses for creating AIGC. Further difficulties arise within the open-source framework, in which actors have white box access to the watermarking scheme. In essence, the issue is that watermarking addresses the problem of provenance of *AIGC*, while the main difficulty is establishing the provenance of *authentic* images.

These asymmetries, where defenders must anticipate all possible attacks while adversaries need only find one successful evasion, motivate the following main question:

Q: Do current AIGC detectors, while being effective under controlled conditions, exhibit significant performance degradation when evaluated against novel generators and simple adversarial perturbations?

This work presents the first systematic empirical evaluation of this hypothesis by benchmarking state-of-the-art text-to-image models against a comprehensive array of commercial AI detection services and simple adversarial perturbations. We make the following contributions:

1) **Benchmark dataset.** We introduce AdversIm, a large-scale benchmark comprising 15630 images across 8 fraud-relevant domains (e.g., receipt manipulation, identity documents) and 2 adversarial perturbation types. The dataset includes 1563 authentic images, 7815 synthetic images from 5 state-of-the-art generators, and 6252 adversarially perturbed variants. Figure 2 shows some sample images from AdversIm.
2) **Detector evaluation.** We evaluate the performance of 7 commercial detection systems on the unperturbed

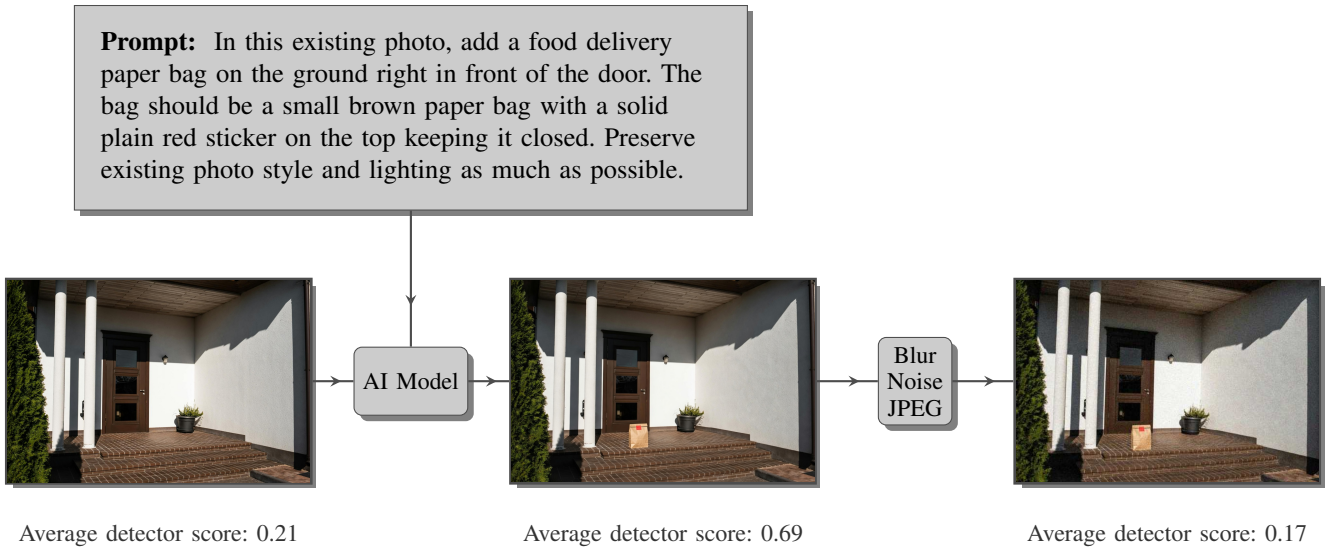

Fig. 1. Experimental workflow. A real image is edited using an AI model prompted with a natural-language instruction. A simple post-processing transformation is then applied. All images are evaluated using an AI-generated image detector.

data. We find that some of the commercial detectors perform well against even the latest generative-AI models (including a pre-release beta version of Grok), while others provide little to no predictive value. Table III and the left column of Figure 4 depict the results of this investigation in more detail. From this investigation, TruthScan [38], Resemble [33], and AI or Not [2] emerge as the most accurate detectors across image domains and models. From the perspective of the models, Grok and Gemini were most successful at bypassing detectors: averaging across detectors, Gemini images were detected 53.5% of the time, while Grok images were detected 54.7% of the time.

3) **Adversarial robustness.** We assess detector robustness under a pair of simple adversarial perturbations (described in more detail in Section III-C). We find that these perturbations dramatically reduce detector effectiveness: the detection rates of the three detectors in Item 2 went from 98%, 90%, 90%, respectively, to 36%, 11%, and 13%. See Section III-C for the full results.

Experiment Artifacts: The dataset and the code used to generate the figures in this article can be found here.[1] The pre-registration for the experiment can be found here.[2] The images of AdversIm and the detector data collected for the study are available here.[3] We also built a website to easily visualize the data used in the experiment [4].

Taken together, our findings indicate that probabilistic detection approaches face inherent limitations that incremental improvements are unlikely to resolve. As generative capabilities advance and adversarial techniques proliferate, we believe the path forward lies not in better classifiers, but in fundamentally different approaches—such as cryptographic provenance—that do not depend on winning an asymmetric arms race.

[1] https://github.com/succinctlabs/ai-benchmark
[2] https://osf.io/dyw7j/overview
[3] Amazon S3 Bucket: `s3://succinct-ai-benchmark`
[4] https://aidetection.succinct.xyz/

II. Background

We quickly review the background necessary to understand the main results in the body of the paper. This is a vast field of active research, and we do not claim to provide an extensive literature review here, choosing instead to provide a general overview. The literature on adversarial perturbations and mitigations against them, in particular, is extensive. The present work determines whether the current state-of-the-art favors the detectors or the generative models, and in particular whether bypassing the detectors is practically feasible with present-day capabilities.

Text-to-image (T2I) or Image Synthesis Generative AI models are neural network architectures trained on the problem of generating or editing images based on a text prompt. The first successful generative image models were trained as generative adversarial networks (GANs) [14], in which two neural nets are trained against each other. More recently, diffusion models have proliferated as a common architecture [34], but other architectures are also being used in practice (e.g. Grok uses an "autoregressive mixture-of-experts network" [1]). The last few years have seen extraordinarily rapid advances in T2I models: the latest generation includes

- Nano Banana Pro/Gemini 3 Pro Image from Google [30]
- Grok Imagine Image from xAI [20]
- OpenAI/ChatGPT Image 1.5 [26]
- Seedream 4.5 from BytePlus [5]
- Qwen-Image-2512 [42] (an open-source model available on Hugging Face).

These models render hyper-realistic, logically consistent images; they struggle much less with rendering human forms, text, and other detailed visual artifacts than previous models did.

Image Editing With AI Models: All of the above-mentioned state-of-the-art models can also take a text-plus-image prompt and edit the image according to the text. The AI edit in Figure 1 demonstrates an example of such capabilities for the Grok Imagine image model. The models are also capable of making more complex edits, for example changing the posture and orientation of its subjects. Image model capabilities of this type can present difficulties for detectors, since they can "piggy-back" off of the realism of the original image and the changes can be very localized. For this reason, AdversIm includes image edits in addition to images that are wholly generated by T2I models.

AI-generated content detectors (AIGC detectors) or synthetic image detectors (SIDs) are, for the purposes of this document, neural net architectures trained on the problem of classifying images that are generated by T2I and/or image editing models. For image models arising from GANs, the detector model of the GAN can be used as a SID.

State-of-the-art research detectors leverage diverse approaches including frequency-domain analysis [24, 29] and data augmentation [40]. Commercial SID services—including TruthScan [38], Reality Defender [31], and others—have emerged to provide enterprise-grade detection capabilities, though their underlying architectures are typically proprietary. Given that the research-grade detectors use convolutional neural nets for their architectures, we hypothesize that the commercial detectors do as well. The commercial detectors provide Application Programming Interfaces (APIs) to which a user can upload images; once the API request is fulfilled, the user obtains among other things the detector's estimated probability that the uploaded image was AIGC.

AI Watermarking is another proposed solution to the AIGC problem. The basic idea of this approach is to embed a visually imperceptible amount of "fingerprinting" into the images generated by an AI model so that the watermark (probabilistically) identifies the image as coming from that model. The watermark is robust against common image transformations, including noise, crop, and rotation [15]. However, recent research suggests that it is possible to cheaply and systematically remove AI watermarks from AIGC [21]. Moreover, our study assumes a threat model where a malicious actor is free to choose the AI model it uses for its attack; under this threat model, if there exists a single effective generative AI model without watermarking capabilities, then AI watermarking is not a sufficient defense. There are further complications when considering the watermarking problem for open-source models, since in this case adversaries have full access to the model and the watermarking scheme.

Adversarial Perturbations are image transformations (e.g. rotate, crop, blur, noise, and compositions thereof) that cause a neural net classifier to misclassify an image despite having correctly classified the untransformed image. The literature on adversarial perturbations goes back to at least 2014 [36]; often, as in [36], the perturbations are designed to be imperceptible to the human eye. An adversarial perturbation can be **black box**, in which case the perturbation is not allowed to depend on internal details (e.g. the weights) of the detector model, or **white box**, in which case the attacker is allowed to know the architecture and weights of the model [13, 36]. It is known that simple image transformations like rotation, noise, and blur pose difficulties for neural net classifiers [27, 39] even when used in a black-box fashion.

There is also a more recent literature [11, 22, 44] on advanced adversarial techniques to bypass neural net classifiers. Often these attacks require some kind of learning procedure either using the detector's weights explicitly in the white-box case or the weights of a proxy for the detector in the black-box setting. In this work we focus on black box attacks of the flavor of [27]: simple image transformations readily available to anyone with basic photo-editing software. Our main contribution in the present work is to demonstrate that these simple perturbations can be enough to bypass a wide range of state-of-the-art commercial detectors.

Mitigations Against Adversarial Perturbations: A number of approaches have been proposed to limit the effectiveness of adversarial perturbations. Data augmentation using adversarially perturbed AIGC can help improve the accuracy of SIDs [40]. Another approach is adversarial training [25], a general framework for improving the robustness of models to adversarial perturbations. This involves training the model to solve a saddle point problem whose inner optimization problem represents finding an adversarial perturbation subject to a bound on its norm (commonly the ℓ_∞-norm) and whose outer optimization problem is to minimize the loss on the dataset subject to the adversarial perturbation.

III. Methodology

A. Dataset Construction

To construct **AdversIm**, we identified 7 domains of real-world interest and assembled relevant authentic images from 8 sources (with at least 85 images in each category, and around 200 in most), as summarized in Table I. [5]

To justify the sample size of 85+ images per class, we formed and pre-registered [23] a hypothesis about the effects of the adversarial perturbations on detector scores and performed a statistical analysis to determine the minimum effect size that could be discriminated by the sample size of 85, targeting Type I error probability $\alpha = 0.05$ and Type II error probability $\beta = 0.20$. We found that 100 samples per class will substantiate an effect size of .6 or greater. We describe the experimental hypothesis and analysis in more detail in Sections III-C and A.

[5] After we pre-registered our experiment, we became aware of Deepfake-Eval-2024 [6], a comprehensive open-source dataset addressed at the deepfake problem for audio, video, and images. Because of time and resource constraints, we were not able to incorporate it into our benchmarking.

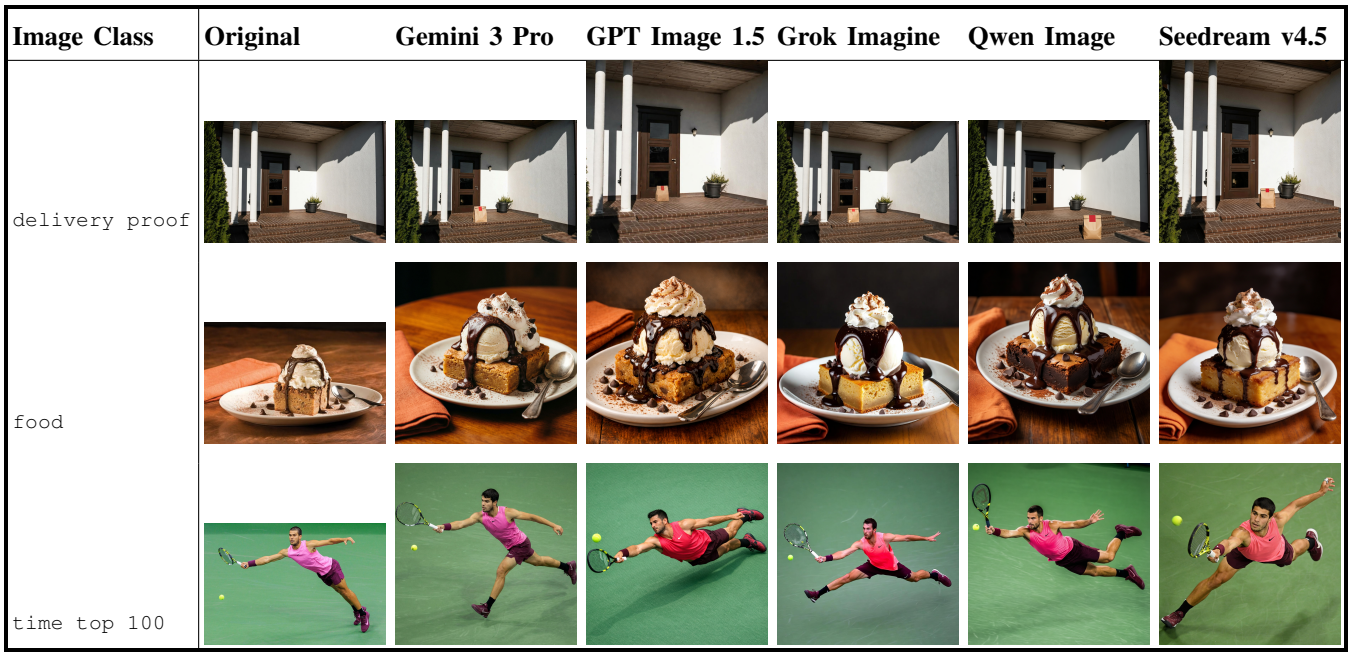

Fig. 2. AI-generated images by image class and model.

Image Class	Source	Description	Edit-type or T2I-type	Description of AI generation	Number of Images
car_add_damage	[4]	Images of automobiles	Edit-type	Add damage to image of undamaged car	196
delivery_proof	[28]	Images of front doors	Edit-type	Add a food delivery package at the front door	157
product	[18]	Low-resolution images of products purchased on e-commerce websites	Edit-type	Add product damage to an image of an undamaged product	289
receipts	[9]	Images of receipts submitted for reimbursement	Edit-type	Double all prices listed on receipt.	183
construction_site	[7]	Images of work at construction sites	T2I-type	N/A	285
food	[43]	Images of food from online restaurant reviews	T2I-type	N/A	199
getty_editorial	[12]	Getty Editorial Images from recent news stories	T2I-type	N/A	169
time_top_100	[37]	Time's Top 100 Photos of 2025	T2I-type	N/A	85

TABLE I
THE 8 IMAGE SOURCES FOR ADVERSIM

We also chose the 5 state-of-the-art AI models listed in the Introduction, of which 4 are commercial and one open-source. For each authentic image of "Edit-type", we fed the image and a text prompt into the models. For each image of "T2I-type", we asked Grok to provide a text description of the image. Then we fed the text descriptions into each of the models, and asked them to produce the image described in the text. Figure 2 displays some examples of images generated by the models. See the Appendix for the prompts used.

B. Detector Evaluation

For the initial phase of the study, we evaluated a set of AIGC detectors against real and unperturbed AIGC and report detector metrics like accuracy, true positive rate (TPR), true negative rate (TNR), and area under receiver operating characteristic curve (AUC-ROC score).

We chose 7 detectors: AI or Not [2], Illuminarty [19], RealScan by RealityDefender [32], Resemble Detect [33], SightEngine [35], TruthScan [38], and Winston AI [41]. We chose all detectors that we could find that fit within our resource constraints.

We interacted with these detectors through their public APIs. The APIs work similarly across detectors: a user uploads an image to the API and the detector reports a score in $[0, 1]$ which represents the detector's estimate of the probability that the image was AI-generated or -modified. Some detectors report further data, for example the estimated

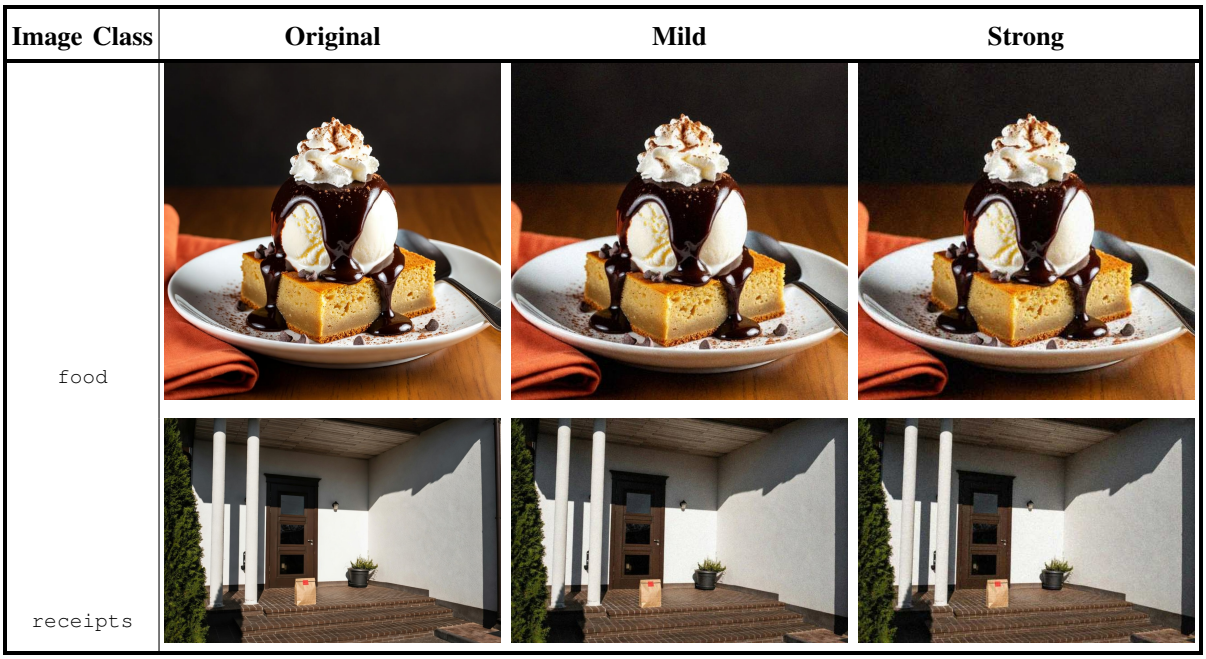

Fig. 3. Adversarial perturbations (mild and strong) applied to Grok Imagine Image Beta generated images by image class.

probability that the image was generated by a specific model such as Grok.

These basic statistics gave us the baseline for the experiment, which we describe below. They also help us to get an estimate of the effectiveness of the AIGC detectors under "normal working conditions". We report the results in Section IV-A.

C. Adversarial Robustness

In this section, we describe the central experimental hypothesis tested in this work. Roughly speaking, the experiment seeks to determine whether it is possible, for any class of fraud, to perturb AI-generated images by *simple* image transformations (e.g. rotation, blur, Gaussian noise, etc., and compositions thereof) so that the perturbed images are not detected by *any* of the detectors. Implicit in the experimental design is a threat model with quite weak assumptions on the capabilities and knowledge of the malicious actor wishing to bypass AIGC detection capabilities: the attacker is ignorant of the particular detector being used against its attack, and has computational capabilities easily attained by most mobile devices.

Let us describe the set of perturbations we wish to consider in the experiment. We study two perturbations, a "mild" and "strong" variant. Both perturbations are a composition of the simple transformations listed in Table II; the Python implementation (with the default values of the parameters) can be found here. Figure 3 gives representative examples of the mild and strong perturbations.

Let C be the set of image classes (i.e., the rows of Table I) for which we want to test the sensitivity of AI-generated content (AIGC) detectors. Because of cost limitations, we choose the two models with the lowest aggregate detection rates in the "Detector Evaluation" phase of the study, and

Image Transformation	Parameter	Value in Weak Perturbation	Value in Strong Perturbation
Rotation	Rotation angle	Default	Default
Blur	Radius	Default	Default
Filmgrain (Film-realistic noise)	Intensity	Default	Default
Gaussian noise	Intensity	7	10
Crop	Coordinates of crop rectangle	Default	Default
JPEG compression	Quality	70	50
Upscale and downscale (Lanczos resampling)	Scaling factor	0.6	0.5

TABLE II

BUILDING BLOCKS FOR THE IMAGE TRANSFORMATIONS USED IN THE EXPERIMENT. THE VALUES MARKED "DEFAULT" ARE SPECIFIED IN THIS PYTHON FUNCTION DECLARATION.

denote this set by M. (The aggregate is taken across all base images in the AdversIm data set.) Let also P denote the set of two adversarial perturbations described above, and D the set of detectors listed in Section III-B. Each $d \in D$ assigns a probability $d(x) \in [0, 1]$ to an image x. For each $m \in M$ and $c \in C$, suppose that we have a (hypothetical) population of all images $I_{m,c}$ generated by the model m for fraud class c. In terms of these symbols, our main experimental hypothesis is:

Hypothesis 1 (H_1): For all $c \in C$, there exist $m \in M$ and $p \in P$ (possibly changing with c) such that

$$\mathbb{E}_{i \in I_{m,c}} [d(p(i)) - d(i)] < 0$$

for all $d \in D$.

As discussed above, this hypothesis is a formalization of a threat model in which an attacker is allowed to choose its perturbation and model based on the particular type of fraud it wishes to commit, but is ignorant of the detector being used against its attack. These seem to be approximately the kind of attacker capabilities one should expect in real-world cases of fraud. For the experiment that tests H_1, we simply measure the detector scores on an AI-generated image before and after a perturbation, and we take the images in AdversIm as the sample from the hypothetical populations $I_{m,c}$. In the Appendix, Section A, we describe how to analyze the type I and type II error probabilities of H_1 using a one-sided t-test for a difference of means with matched pairs. In brief, we analyze the error probabilities in terms of those of the hypotheses $H_{c,m,p,d}$, which are similar to H_1, except that c, m, p, d are fixed.

IV. Results

A. Detector Evaluation

Table III and the leftmost column of Figure 4 report the results of detector evaluation on the base data set. (Figure 4 only reports the results for the two most successful generative models–Grok and Gemini–in a visual format.)

Table III reports, for each image class and detector, a true negative rate (the fraction of original/real images given a score less than 0.5 by the detector), a true positive/detection rate (the fraction of AI-generated or edited images given a score greater than 0.5 by the detector), overall accuracy, and a ROC-AUC score. We note that the dataset contains 5 AI-generated images for every real image (because we studied 5 AI models), so the aggregate scores (accuracy and AUC-ROC) should be taken as informative but only in a rough sense, since e.g., the real-world distribution of images doesn't necessarily contain 5 times as many AI-generated images as real ones.

We highlight the following observations:

- All detectors have a relatively low false positive rate $1 - TNR$, i.e. they are relatively unlikely to classify real images as AI-generated.
- There is a wide variability in the detection rates (true positive rates) between detectors. TruthScan, Resemble, and AI or Not typically had the best detection rates. In particular, TruthScan worked well for all models and all image classes.
- Some detectors performed much better on the "T2I-type" image categories than the "edit" categories. Illuminarty most exemplified this pattern, but SightEngine also falls into this category.
- The detectors seem to struggle the most on the `product` image category. This is probably because the images in this category are relatively low-resolution and the generated images are of "edit"-type. They struggle least on the "T2I"-type images, and in particular on the `food` image category.

We note also the following overall ranking of models by overall detection rates across all detectors and images (low detection rate means a more "deceptive" model):

1) Gemini: 53.5%
2) Grok: 54.7%
3) Seedream: 60.8%
4) GPT: 63.4%
5) Qwen: 78.0%

Furthermore, we note the following ranking of AIGC detection rates across all models and images:

1) TruthScan: 95.3%
2) AI or Not: 93.8%
3) Resemble: 81.4%
4) SightEngine: 48.9%
5) Illuminarty: 40.7%
6) RealityDefender: 38.3%
7) Winston AI: 36.2%

We note for both rankings that these are only meant to be crude summary statistics; in particular, the relative class-by-class sample sizes are not tuned to be representative of any particular use-case. For this reason, the class-by-class statistics can give more informative and granular data. Moreover, both models and detectors varied in terms of pricing, which necessitates further caution when assessing the above numbers against each other: it is possible that some of the detectors are more effective but are more expensive to run. Since in this experiment we are mostly concerned with state-of-the-art capabilities, we did not consider relative costs when evaluating the models and detectors.

If we were to stop the study at this point, we would conclude that there exists at least one detector which can effectively detect (unperturbed) AIGC—including content of "edit"-type and "T2I"-type—from many different models and in many different use cases. However, as we will see below, this conclusion changes dramatically once one allows for adversarial images.

B. Adversarial Robustness

Figure 4 illustrates the effects of the perturbations on the detectors' performance. In this figure, the two most successful models from the initial detector evaluation (Grok and Gemini) are each assigned a row of heatmaps. The leftmost heatmap in each row corresponds to the unperturbed (but still AI-generated) images, the middle heatmap to the mildly perturbed images, and the rightmost to the strongly perturbed images. Each heatmap reports detection rates (true positive rate, assuming a threshold of 0.5) for each detector across the 8 categories of images. It also reports the aggregate score of each detector on the entire dataset. A similar figure in the appendix (Figure 6) reports the effect sizes and p-values organized by model, category, and detector. We defer fuller discussion of the effect-size and p-value analysis to the appendix, but we highlight a few key aspects of the figures here:

1) The mild perturbation of the Grok images was enough to dramatically reduce overall detector performance

Detector	car_add_damage				construction_site				delivery_proof				food			
	TPR	TNR	AUC	Acc	TPR	TNR	AUC	Acc	TPR	TNR	AUC	Acc	TPR	TNR	AUC	Acc
AI or Not	96.8	95.4	99.5	96.6	99.9	95.4	100.0	99.1	98.0	96.2	99.5	97.7	100.0	89.9	99.4	98.3
Illuminarty	25.6	93.4	67.4	36.9	23.5	96.1	90.7	35.6	20.6	91.7	62.1	32.5	79.1	86.9	91.8	80.4
RealityDefender	19.1	93.9	51.4	31.5	49.8	97.9	85.8	57.8	23.2	98.1	65.6	35.7	47.6	100.0	89.4	56.4
Resemble	86.3	99.5	98.0	88.5	90.5	85.3	94.5	89.6	92.9	95.5	97.8	93.3	69.7	97.0	91.4	74.3
SightEngine	30.0	99.5	74.3	41.6	65.9	99.6	93.8	71.5	23.1	96.2	68.5	35.2	79.0	99.5	95.9	82.4
TruthScan	97.1	99.0	99.9	97.4	95.9	98.2	98.9	96.3	88.4	96.8	98.8	89.8	96.3	95.5	99.0	96.1
Winston AI	5.1	97.4	52.9	20.5	50.5	96.8	86.3	58.2	34.9	63.1	48.3	39.6	85.9	77.4	90.6	84.5

Detector	getty_editorial				product				receipts				time_top_100			
	TPR	TNR	AUC	Acc	TPR	TNR	AUC	Acc	TPR	TNR	AUC	Acc	TPR	TNR	AUC	Acc
AI or Not	100.0	97.6	100.0	99.6	86.7	97.6	98.8	88.5	73.6	100.0	96.9	78.0	99.5	98.8	99.9	99.4
Illuminarty	69.5	95.9	91.8	73.9	37.2	100.0	87.2	47.6	20.5	94.0	70.4	32.8	78.4	90.6	91.7	80.4
RealityDefender	72.5	97.0	91.3	76.6	30.9	96.9	37.0	41.9	3.9	100.0	50.2	19.9	80.5	95.3	94.2	82.9
Resemble	78.9	98.2	93.4	82.1	68.9	100.0	92.3	74.0	84.6	98.9	97.5	87.0	85.4	81.2	91.8	84.7
SightEngine	70.9	99.4	95.0	75.6	37.2	99.7	80.2	47.6	16.1	98.4	69.5	29.8	79.5	92.9	94.0	81.8
TruthScan	98.2	100.0	100.0	98.5	91.5	98.6	99.1	92.7	99.8	95.6	100.0	99.1	96.9	100.0	99.9	97.5
Winston AI	46.5	92.3	82.3	54.1	19.8	89.3	55.2	31.4	0.3	100.0	51.8	16.9	59.1	82.4	74.5	62.9

TABLE III

AI DETECTION PERFORMANCE ACROSS IMAGE CLASSES, AGGREGATED OVER GENERATIVE MODELS

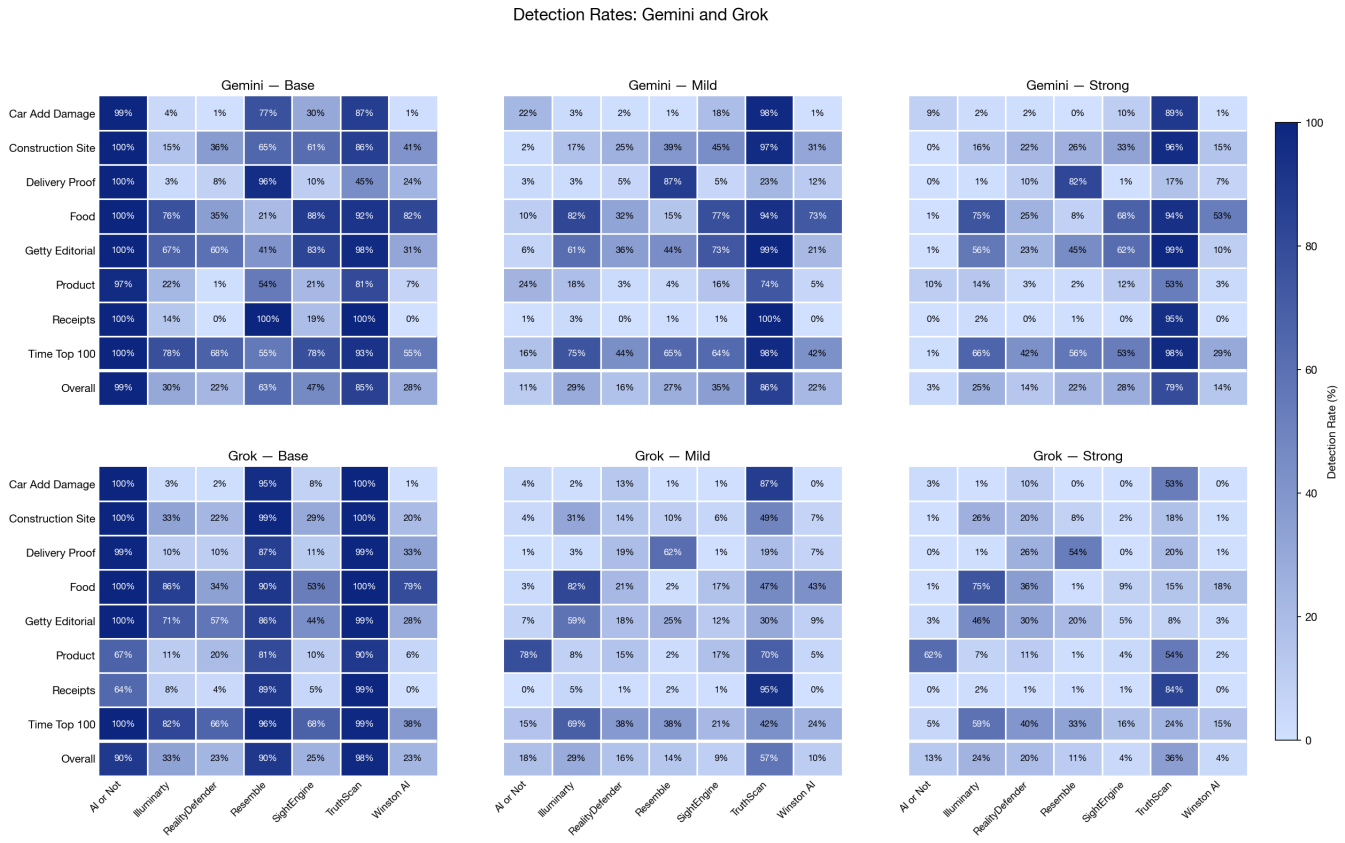

Fig. 4. The effects of the mild and strong perturbations on detector scores by category and model.

for all detectors. Except for some isolated detector/category pairs, the overall detection rates were low enough not to be practically useful (even with the *mild* perturbation).

2) The results for the perturbations of Gemini images were more mixed: TruthScan's detection rates were degraded but it remained usable even under the strong perturbation.
3) For all but the `car_add_damage` class, it was possible to find a model and perturbation that decreased *all* detectors' average scores on the images in that class. For the `car_add_damage` class, there was a consistent negative effect (the scores increased for all perturbations and models) for RealityDefender, though the baseline detection rate for RealityDefender was extremely low to begin with: 2 percent.
4) The effect sizes were generally largest and the p-values smallest for the three detectors that were most effective on the base AI-generated images. The other detectors typically had much lower detection rates to begin with, so the effects were less pronounced.
5) TruthScan remains relatively good at detecting the receipt images even with the strong perturbation and even on Grok-generated images, while Illuminarty remains relatively effective against the food data class. Nevertheless, the effect sizes are still 1.5 and .5, respectively, with $p < 0.001$.

V. Discussion

In the Introduction, we posed the question: how robust are AI-based AIGC detectors against adversarial attacks? The results we reported in the previous section constitute a very strong case for the answer: "not very". In this section, we explore some further evidence for this answer, and discuss potential solutions.

One possible way to address weaknesses of any one individual detector is to ensemble detectors in some way, for example by taking a majority vote among detectors (or more generally taking the average score assigned by detectors) or by considering an image to be AI-generated if at least one of the detectors does (max-rule combination). To assess the effectiveness of the perturbation against ensembles of detectors, we analyzed AUC-ROC scores for the max, min, and mean detectors on the images generated by the Grok image model; we chose the AUC-ROC score because this gives some measure of the tradeoffs between false positives and true positives.

As a quick reminder: for detectors that only output a binary classification, the max-rule detector corresponds to the or-rule: the combined detector "fires" if at least one of the individual ones does. Similarly, the min-rule model corresponds to the and-rule, i.e. to a unanimous vote of the individual classifiers; and the mean-rule with threshold .5 corresponds to majority vote of the individual classifiers. The results of the analysis for these three ensembled detectors are presented in Figure 5. The graphs show that the max and average detectors are quite effective on the unperturbed data. On the other hand, the min-rule ensemble is not very effective; it turns out that this is because the ineffective detectors "hold back" the min rule ensemble (the curves are more interpretable if only the three best models are used, though for brevity we omit a fuller discussion of this case). Furthermore, the perturbations are, at the aggregate level (across image classes), still effective on all three ensembles: the min model becomes worse than random chance after even the mild perturbation.

It is also possible to consider more complex ensembles by, e.g., taking a weighted average of the detectors where the weights are learned. However, the preliminary analysis of the simpler ensembles and the fact that the perturbations worked universally across all detectors suggest that even the more complex ensembles will suffer from the same attack.

More generally, the universality of this perturbation suggests a fundamental limitation of AI-based solutions to the problem of AIGC. Our analysis of the problem is in some ways inspired by the cryptographic mindset: in cryptography, solutions are never tailored to an average-case scenario and it is always important to keep in mind what happens when a determined (if bounded) adversary is given access to the system. For a problem as central to society as that posed by AIGC, it is necessary to find a truly robust solution with guarantees of the sort provided by cryptography. Instead of probabilistically detecting what is fake, we should *prove* what is real.

References

[1] X AI. *Grok Image Generation Release*. Last accessed 26 January 2026. 2024. URL: https://x.ai/news/grok-image-generation-release.

[2] AI or Not. *AI or Not: AI-Generated Content Detector*. https://www.aiornot.com/. Accessed: 2025-01-30. 2025.

[3] Anish Athalye et al. *Synthesizing Robust Adversarial Examples*. 2018. arXiv: 1707.07397 [cs.CV]. URL: https://arxiv.org/abs/1707.07397.

[4] Auto.dev. *Auto.dev: Simple, powerful, and easy to use APIs for automotive.* https://www.auto.dev. Accessed: 2025-01-30. 2025.

[5] BytePlus. *BytePlus Unveils Seedream 4.5: Precision-Focused Upgrade Delivering Sharper Visuals, Smarter Control, and 4K Creative Consistency*. https://www.byteplus.com/en/blog/seedream4-5. Accessed: 2026-01-29. 2025.

[6] Nuria Alina Chandra et al. *Deepfake-Eval-2024: A Multi-Modal In-the-Wild Benchmark of Deepfakes Circulated in 2024*. 2025. arXiv: 2503.02857 [cs.CV]. URL: https://arxiv.org/abs/2503.02857.

[7] Xuezheng Chen and Zhengbo Zou. *Are Large Pre-trained Vision Language Models Effective Construction Safety Inspectors?* 2025. arXiv: 2508.11011 [cs.CV]. URL: https://arxiv.org/abs/2508.11011.

[8] Cristina Criddle. "'Do not trust your eyes': AI generates surge in expense fraud". In: *Financial Times* (Oct. 2025). URL: https://www.ft.com/content/0849f8fe-2674-4eae-a134-587340829a58 (visited on 01/28/2026).

[9] ExpressExpense. *ExpressExpense SRD: Sample Receipt Dataset for OCR and Machine Learning*. https://expressexpense.com/blog/free-receipt-images-ocr-machine-learning-dataset/. Licensed under MIT License. Accessed: 2025-01-30. 2020.

[10] Franz Faul et al. "Statistical power analyses using G*Power 3.1: Tests for correlation and regression analyses". In: *Behavior Research Methods* 41.4 (2009), pp. 1149–1160. DOI: 10.3758/BRM.41.4.1149.

[11] Chiara Galdi et al. "2D-Malafide: Adversarial Attacks Against Face Deepfake Detection Systems". In: *2024 International Conference of the Biometrics Special Interest Group (BIOSIG)*. IEEE, Sept. 2024, pp. 1–7. DOI: 10.1109/biosig61931.2024.10786754.

[12] Getty Images. *Getty Images Editorial Images Collection*. https://www.gettyimages.com/editorial-images. Accessed: 2025-01-30. 2025.

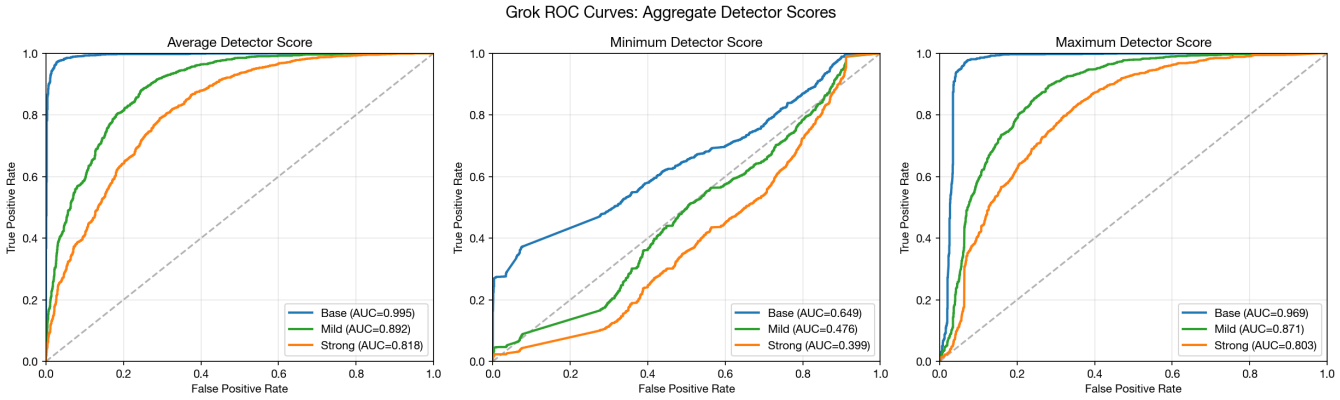

Fig. 5. ROC curves for aggregate detectors.

[13] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. *Explaining and Harnessing Adversarial Examples*. arXiv:1412.6572 [stat]. Mar. 2015. DOI: 10.48550/arXiv.1412.6572. URL: http://arxiv.org/abs/1412.6572 (visited on 01/27/2026).

[14] Ian J. Goodfellow et al. *Generative Adversarial Networks*. Version Number: 1. 2014. DOI: 10.48550/ARXIV.1406.2661. URL: https://arxiv.org/abs/1406.2661 (visited on 01/27/2026).

[15] Google DeepMind. *Identifying AI-Generated Images with SynthID*. Google DeepMind Blog. Accessed: 2026-02-11. Aug. 2023. URL: https://deepmind.google/blog/identifying-ai-generated-images-with-synthid/.

[16] Anthony Ha. *Doordash Says it Banned Driver Who Seemingly Faked a Delivery Using AI*. Jan. 2026. URL: https://techcrunch.com/2026/01/04/doordash-says-it-banned-driver-who-seemingly-faked-a-delivery-using-ai/ (visited on 01/29/2026).

[17] Christopher Harris. *Georgia Rep. Mike Collins' campaign uses AI-generated deepfake of Senator Jon Ossoff in tight Senate showdown*. Online News. Nov. 2025. URL: https://www.cbsnews.com/atlanta/news/georgia-rep-mike-collins-campaign-uses-ai-generated-deepfake-of-senator-jon-ossoff-in-tight-senate-showdown/ (visited on 01/29/2026).

[18] Yupeng Hou et al. "Bridging Language and Items for Retrieval and Recommendation". In: *arXiv preprint arXiv:2403.03952* (2024).

[19] Illuminarty. *Illuminarty: AI-Generated Image Detection*. https://app.illuminarty.ai/. Accessed: 2025-01-30. 2025.

[20] *Image Generations and Edits*. API Documentation. URL: https://docs.x.ai/docs/guides/image-generations.

[21] Andre Kassis and Urs Hengartner. "UnMarker: A Universal Attack on Defensive Image Watermarking". In: *2025 IEEE Symposium on Security and Privacy (SP)*. IEEE, May 2025, pp. 2602–2620. DOI: 10.1109/sp61157.2025.00005. URL: http://dx.doi.org/10.1109/SP61157.2025.00005.

[22] Andre Kassis, Urs Hengartner, and Yaoliang Yu. *DiffBreak: Is Diffusion-Based Purification Robust?* 2024. DOI: 10.48550/ARXIV.2411.16598.

[23] Succinct Labs. *Preregistration. Seeing Is No Longer Believing: Benchmarking Synthetic Image Detection*. Feb. 2026. DOI: 10.17605/OSF.IO/DYW7J. URL: osf.io/dyw7j.

[24] Jun Li et al. "Optimized Frequency Collaborative Strategy Drives AI Image Detection". In: *IEEE Internet of Things Journal* 12.11 (June 2025), pp. 16192–16203. ISSN: 2372-2541. DOI: 10.1109/jiot.2025.3531053.

[25] Aleksander Madry et al. *Towards Deep Learning Models Resistant to Adversarial Attacks*. arXiv:1706.06083 [stat]. Sept. 2019. DOI: 10.48550/arXiv.1706.06083. URL: http://arxiv.org/abs/1706.06083 (visited on 01/27/2026).

[26] OpenAI. *The new ChatGPT Images is here*. https://openai.com/index/new-chatgpt-images-is-here/. Accessed: 2026-01-29. 2025.

[27] Kexin Pei et al. *Towards Practical Verification of Machine Learning: The Case of Computer Vision Systems*. 2017. arXiv: 1712.01785 [cs.CR]. URL: https://arxiv.org/abs/1712.01785.

[28] Pexels. *Pexels: Free Stock Photos and Videos*. https://www.pexels.com. Accessed: 2025-01-30. 2025.

[29] Orazio Pontorno, Luca Guarnera, and Sebastiano Battiato. "On the Exploitation of DCT-Traces in the Generative-AI Domain". In: *2024 IEEE International Conference on Image Processing (ICIP)*. IEEE, Oct. 2024, pp. 3806–3812. DOI: 10.1109/icip51287.2024.10648013.

[30] Naina Raisinghani. *Introducing Nano Banana Pro*. Nov. 2025. URL: https://blog.google/innovation-and-ai/products/nano-banana-pro/ (visited on 01/29/2026).

[31] Reality Defender. *Deepfake Detection Platform*. https://realitydefender.com. Accessed: 2026-01-27. 2024.

[32] Reality Defender. *Reality Defender: Deepfake and AI-Generated Media Detection*. https://www.realitydefender.com/. Accessed: 2025-01-30. 2025.

[33] Resemble AI. *Resemble Detect: AI-Generated Audio and Deepfake Detection*. https://www.resemble.ai/detect. Accessed: 2025-01-30. 2025.

[34] Robin Rombach et al. "High-Resolution Image Synthesis with Latent Diffusion Models". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022. URL: https://github.com/CompVis/latent-diffusionhttps://arxiv.org/abs/2112.10752.

[35] Sightengine. *Sightengine: Detect AI-Generated Images*. https://sightengine.com/detect-ai-generated-images. Accessed: 2025-01-30. 2025.

[36] Christian Szegedy et al. "Intriguing properties of neural networks". In: *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2014. URL: http://arxiv.org/abs/1312.6199.

[37] TIME Magazine. *TIME's Top 100 Photos of 2025*. https://time.com/7336112/top-100-photos-2025/. Accessed: 2025-01-30. Nov. 2025.

[38] TruthScan. *TruthScan: AI-Generated Content Detector*. https://truthscan.com/. Accessed: 2025-01-30. 2025.

[39] Ge Wang et al. "The Detection Optimization of Low-Quality Fake Face Images: Feature Enhancement and Noise Suppression Strategies". In: *Applied Sciences* 15.13 (June 2025), p. 7325. ISSN: 2076-3417. DOI: 10.3390/app15137325.

[40] Sheng-Yu Wang et al. "CNN-Generated Images Are Surprisingly Easy to Spot... for Now". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 8695–8704. DOI: 10.1109/CVPR42600.2020.00872.

[41] Winston AI. *Winston AI Content Detector*. https://gowinston.ai/ai-content-detector/. Accessed: 2025-01-30. 2025.

[42] Chenfei Wu et al. *Qwen-Image Technical Report*. 2025. arXiv: 2508.02324 [cs.CV]. URL: https://arxiv.org/abs/2508.02324.

[43] Yelp Inc. *Yelp Open Dataset*. https://business.yelp.com/data/resources/open-dataset/. Accessed: 2025-01-30. 2025.

[44] Ziyin Zhou et al. "StealthDiffusion: Towards Evading Diffusion Forensic Detection through Diffusion Model". In: *Proceedings of the 32nd ACM International Conference on Multimedia*. MM '24. ACM, Oct. 2024, pp. 3627–3636. DOI: 10.1145/3664647.3681535.

APPENDIX

A. Statistical Power Analysis

In this section, we provide a more formal analysis of the experimental hypothesis and identify the sample size needed to substantiate a range of effect sizes to our desired accuracy.

Given a target maximum probability α of type I error, a target maximum probability β of type II error, and a target minimum effect size z, we would like to estimate the size n of a sample needed to determine whether hypothesis H_1 is true to that level of confidence and power.

If there were no quantifiers in the statement of H_1, this would be a well-established problem: the difference of means problem between two dependent samples (matched pairs). Standard tools exist to analyze the sample size needed to substantiate a difference of means between two data sets [10]. One reports the desired α, β and effect size (mean of the difference between the two samples–perturbed and unperturbed–normalized by the standard deviation of the difference) and obtains the sample size needed to demonstrate that the means of the two distributions are distinct.

The only reason that the present analysis is somewhat different is the presence of quantifiers in H_1. However, we now show that we can analyze H_1 by reducing it to a number of individual difference-of-means hypotheses. To this end, fix $c \in C, m \in M, p \in P, d \in D$, and consider the hypothesis

$$H_{c,m,p,d}: \quad \mathbb{E}_{i \in I_{m,c}}[d(p(i)) - d(i)] < 0.$$

It is evident that the hypothesis H_1 is a logical expression in terms of the individual $H_{c,m,p,d}$ hypotheses, so it can be substantiated or rejected by running the individual experiments.

We seek to understand how the type I and type II error probabilities of H_1 can be estimated from the corresponding error probabilities for the individual hypotheses $H_{c,m,p,d}$, which we denote by $\alpha_{c,m,p,d}$ and $\beta_{c,m,p,d}$, respectively. Intuitively speaking, the ability to search combinations of AI model and perturbation to confirm H_1 makes the probability of type I error higher for H_1 than for the individual hypotheses, for the same reason that p-hacking is a concerning phenomenon in experimental sciences. In a similar vein, the need to bypass all detectors, and to do so for each class, increases the probability of type II error for H_1. A more precise and quantitative analysis is given by the following Claim.

Claim 1: If

$$\alpha_{H_{c,m,p,d}} \leq \frac{\alpha}{|M||P|}, \quad \beta_{H_{c,m,p,d}} \leq \frac{\beta}{|C||D|}$$

for all $(c, m, p, d) \in C \times M \times P \times D$, then

$$\alpha_{H_1} \leq \alpha, \quad \beta_{H_1} \leq \beta.$$

Proof: Each "for all" statement in H_1 represents a conjunction of individual hypotheses, while a "there exists" statement represents a disjunction. Therefore,

$$H_1 = \bigwedge_{c \in C} \bigvee_{\substack{m \in M \\ p \in P}} \bigwedge_{d \in D} H_{c,m,p,d}.$$

Let us recall that, given two hypotheses H and H' with type I and type II error probabilities $\alpha_H, \alpha_{H'}, \beta_H, \beta_{H'}$, the conjunction $H \wedge H'$ of the two hypotheses has the properties:

$$\alpha_{H \wedge H'} \leq \min(\alpha_H, \alpha_{H'}), \quad \beta_{H \wedge H'} \leq \beta_H + \beta_{H'},$$

since we mistakenly reject the null of $H \wedge H'$ only if we mistakenly reject the null of both H and H', while we mistakenly fail to reject the null of $H \wedge H'$ if we mistakenly fail to reject the null of either H or H'. Similarly, the disjunction $H \vee H'$ satisfies

$$\alpha_{H \vee H'} \leq \alpha_H + \alpha_{H'}, \quad \beta_{H \vee H'} \leq \min(\beta_H, \beta_{H'}).$$

Therefore,

$$\alpha_{H_1} \leq \min_{c \in C} \left(\sum_{\substack{m \in M \\ p \in P}} \min_{d \in D} \alpha_{H_{c,m,p,d}} \right) \leq \sum_{\substack{m \in M \\ p \in P}} \frac{\alpha}{|M||P|} \leq \alpha.$$

A similar analysis applies for the β probability. ∎

Hence, we can reduce the sample size analysis of H_1 to an analysis of each individual $H_{c,m,p,d}$, as long as we adjust the type I and type II error probabilities accordingly. For the concrete case $|C| = 8$, $|D| = 7$, $|P| = 2$, $|M| = 2$, the adjusted error bounds from the Claim give $\alpha_{H_{c,m,p,d}} \leq \alpha/4$ and $\beta_{H_{c,m,p,d}} \leq \beta/56$. Using standard power analysis tools [10], 100 samples per class are enough to substantiate each $H_{c,m,p,d}$ as long as the effect size is at least ≈ 0.55.

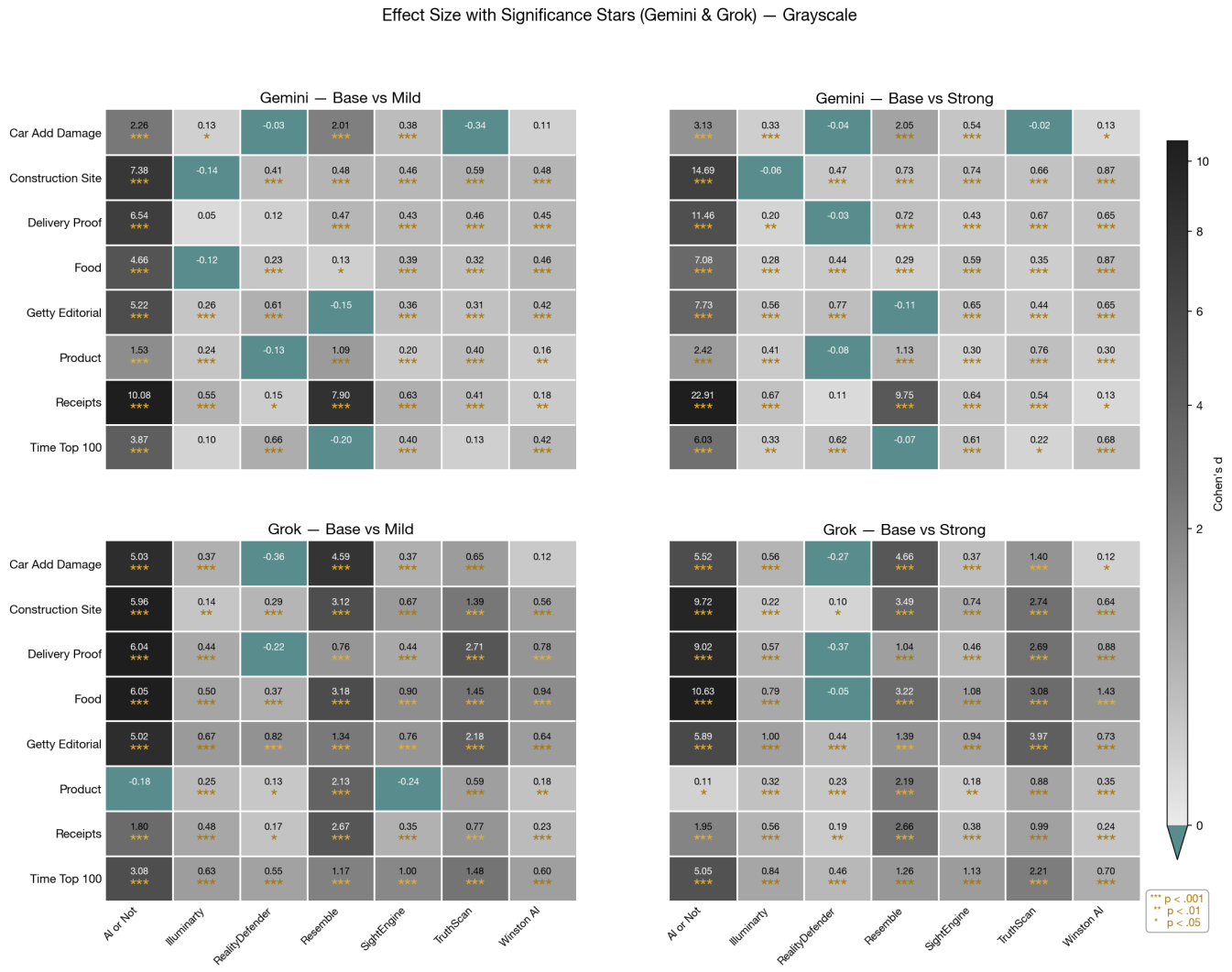

Fig. 6. Effect sizes and p-values for the adversarial perturbations.

B. Statistical Analysis

Figure 6 reports the Cohen's d effect sizes and statistical significance levels for each combination of model (Gemini and Grok), perturbation strength (mild and strong), image category, and detector. Each cell displays the effect size, with significance indicated by a number of stars (***: $p < 0.001$, **: $p < 0.01$, *: $p < 0.05$).

Several patterns are apparent from the figure:

- The effect sizes are generally positive across all four model-perturbation panels, confirming that the perturbations systematically reduce detector scores. The strong perturbation generally produces larger effect sizes than the mild perturbation, as expected.
- The largest effect sizes (frequently exceeding 5.0) and most significant p-values tend to occur for the three most effective detectors identified in Section IV-A—TruthScan, AI or Not, and Resemble—since these detectors had the highest baseline detection rates and therefore the most room for degradation.
- For Grok, the mild perturbation alone produces large, highly significant effects for nearly all detector-category pairs. For Gemini, the results are more mixed: TruthScan retains moderate effectiveness under the mild perturbation for several categories, though the strong perturbation generally produces effects even for TruthScan.
- A notable exception is the `car_add_damage` category for RealityDefender, where the effect sizes are negative (i.e., all perturbations of all models slightly increased detector scores). However, this detector's baseline detection rate for this category was extremely low (2%), so this anomaly has little practical significance.

These results provide strong statistical support for Hypothesis H_1: for each image category, there exists a model-perturbation combination that significantly reduces all detectors' scores, with the exception noted above.

C. Prompts

For the edit-type image categories, we used the following fixed prompts for each category:

- `car_add_damage`: "Add realistic moderate body damage to this car—major scratches or moderate dents on the body panels. Make the damage look natural and real, as if from a minor collision. Preserve existing photo style and lighting as much as possible."
- `delivery_proof`: "In this existing photo, add a food delivery paper bag on the ground right in front of the door. The bag should be a small brown paper bag with a solid plain red sticker on the top keeping it closed. Preserve existing photo style and lighting as much as possible."
- `product`: "Edit this photo so this product looks like it arrived from the seller with a realistic issue that a customer would complain about. The issue can be either minor shipping damage (scuff/tear/ rip/dent, paint chip, or a small crack) OR a manufacturing defect (uneven paint finish, misaligned seam, or a surface blemish). It still has to look like a real photo."
- `receipts`: "Edit this receipt image to exactly double the total amount and all intermediate monetary values, including line item prices, subtotal, tax amount, total, and any other amount necessary. Keep everything else in the photo exactly the same."

For the T2I-type image categories (`construction_site`, `food`, `getty_editorial`, `time_top_100`), we used a two-step process. First, we prompted a vision-language model to produce a detailed textual description of each authentic image. Specifically, we used the following system prompt:

> "You are an expert at describing images in precise detail for image generation AI models. Analyze this photograph and write a detailed prompt that could be used to recreate this exact image. Focus on: main subject and its characteristics; setting/environment; lighting conditions; camera angle and perspective; colors, textures, and materials; any distinctive features or details. Write a single detailed paragraph that captures all visual elements. Be specific and precise. Use 1024 characters or less. Respond only with the description, no other text."

The resulting image descriptions were then fed as text prompts to each of the five generative models to produce the synthetic images.