MirrorCheck: Efficient Adversarial Defense for Vision-Language Models

Samar Fares[1*] Klea Ziu[1*] Toluwani Aremu[1*] Nikita Durasov[2] Martin Takáč[1]
Pascal Fua[3] Ivan Laptev[1] Karthik Nandakumar[1,4]
[1]Mohamed Bin Zayed University of Artificial Intelligence (MBZUAI)
[2]NVIDIA [3]École Polytechnique Fédérale de Lausanne (EPFL)
[4]University of Michigan

Abstract

Vision-Language Models (VLMs) are increasingly susceptible to sophisticated adversarial attacks, including adaptive strategies specifically designed to bypass existing defenses. To address this vulnerability, we propose `MirrorCheck`*, a robust and model-agnostic detection framework that operates effectively in both unimodal and multimodal settings.* `MirrorCheck` *leverages Text-to-Image (T2I) models to regenerate visual content from captions produced by the target model and assesses semantic consistency by comparing feature-space embeddings between the original and synthesized images. To enhance robustness against adaptive attacks,* `MirrorCheck` *introduces a stochastic defense strategy that randomly selects T2I generators and image encoders from a diverse model zoo. Additionally, we incorporate a novel One-Time-Use (OTU) perturbation applied to the selected encoder embeddings, regulated by a scaling factor, which decreases the effectiveness of adaptive attacks. Extensive experiments across multiple threat scenarios demonstrate that* `MirrorCheck` *consistently outperforms baseline methods, and maintains its utility even under strong adaptive adversarial conditions.*

1. Introduction

Vision-Language Models (VLMs) have emerged as powerful tools at the intersection of computer vision (CV) and natural language processing (NLP), enabling machines to reason jointly across modalities and deliver state-of-the-art performance in tasks such as image captioning (IC), visual question answering (VQA), and image text retrieval [1–5]. However, alongside their impressive capabilities comes an increased susceptibility to adversarial attacks, maliciously crafted inputs that cause models to produce incorrect or misleading outputs with imperceptible perturbations [6–8].

Various strategies, such as detectors [9, 10], purifiers [11, 12], adversarial training [13], and certified defenses [14], have been proposed to defend against adversarial threats. However, these methods are specifically tailored for image classification tasks, sometimes requiring expensive retraining and task-specific tuning, while remaining vulnerable to adaptive attacks [15]. While recent efforts [16–19] have explored improving the adversarial robustness of VLMs, these approaches do not provide empirical guarantees against white-box adaptive attacks.

To address this gap, we introduce `MirrorCheck`, a novel, model-agnostic adversarial detection framework for VLMs. Specifically, we leverage Text-to-Image (T2I) models to regenerate images from captions produced by the potentially compromised model, and compare the original and generated images in the embedding space using randomly chosen image encoders. A lower similarity score indicates a likely adversarial sample. We propose two variants of our method: **Vanilla**, which establishes the core detection pipeline using T2I generation and embedding comparison; and **Stochastic**, which leverages randomized model choice and weight transformations for robustness against adaptive attacks. This layered stochasticity increases the search space for attackers, rendering white-box adaptive attacks computationally intractable. To summarize: **(i)** We present `MirrorCheck`, a framework for detecting adversarial samples in VLMs. `MirrorCheck` is a plug-and-play approach which doesn't require training and is model-agnostic. **(ii)** We further propose a stochastic extension of `MirrorCheck` that introduces randomness and controlled perturbations to thwart adaptive attacks. **(iii)** Extensive empirical evaluations across various attack settings demonstrates that `MirrorCheck` outperforms baselines and maintains strong performance under adaptive threat models. Results also reveals that `MirrorCheck` can also generalize to unimodal tasks.

2. Related Work

In this section, we briefly review a few attacks and defenses relevant to our study.

*Equal contribution

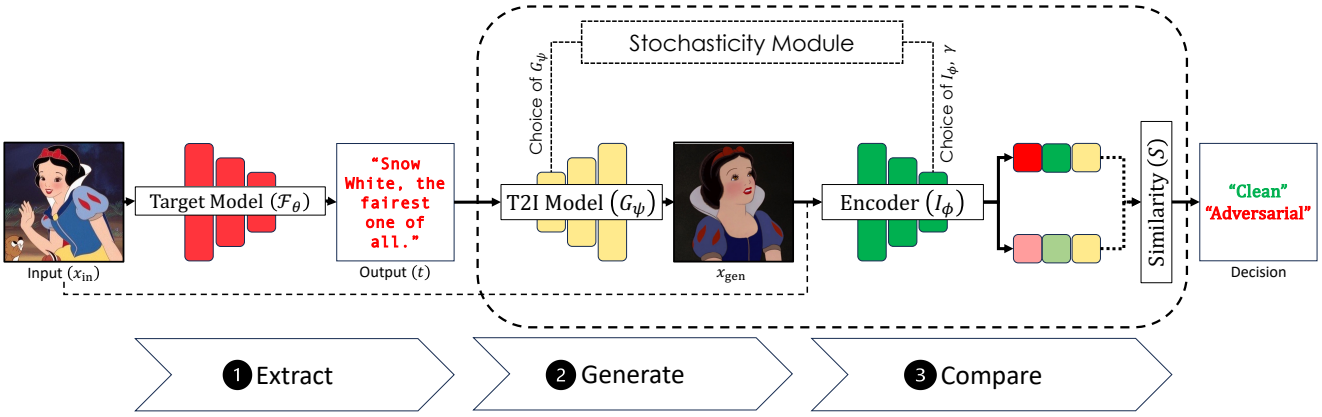

Figure 1. **MirrorCheck framework:** (1) **Extract**: An input image is passed through the victim model to generate an output (caption/description/answer/classification). (2) **Generate**: The output is fed to a randomly selected T2I diffusion model, returning a generated image x_{gen}. (3) **Compare**: We extract and compare feature embeddings from both original x_{in} and generated x_{gen} images using randomly selected and uniquely perturbed image encoders. Significant embedding discrepancies indicate potential adversarial attacks.

2.1. Adversarial Attacks

Adversarial attacks exploit model vulnerabilities through perturbations that cause misclassification or targeted misbehavior. Early research focused on unimodal architectures, particularly CNNs for image classification [13, 20–22]. These attacks are categorized into white-box settings (providing full model access and enabling gradient-based methods like FGSM [20] and PGD [13]) and black-box settings (relying on transferability or query-based methods [23]). Recent advances have extended attacks to multimodal systems, particularly vision-language models (VLMs). AttackVLM [7] introduces transfer-based and query-based strategies targeting VLMs in black-box scenarios, while VLATTACK [6] and Attack-Bard [24] combine image and text perturbations. These attacks exploit the architectural complexity of VLMs, where perturbations can impact both visual and textual modalities, potentially making them more vulnerable than their unimodal counterparts [25–28].

2.2. Adversarial Defenses

Traditional adversarial defenses for unimodal tasks include detection methods [10, 29, 30], purification techniques [11, 12, 31], adversarial training [13, 32], and certified defenses [14, 33]. However, these approaches face critical limitations when applied to VLMs, as they operate on single modalities and cannot account for complex visual-linguistic interactions that adversaries exploit. Furthermore, conventional defenses are vulnerable to adaptive attacks where adversaries with white-box access systematically bypass protection mechanisms [15, 34, 35].

More recently, a new paradigm of training-free defense strategies [18, 19, 36, 37] have been proposed to safeguard VLMs, validating the paradigm's effectiveness. Our work follows this paradigm and introduces a framework that leverages T2I models and employs stochasticity to resist adaptive attacks. We provide detailed descriptions of specific attack methods and baseline defenses in Appendix A.

3. Method

Let $\mathcal{F}_\theta(x_{\text{in}}; p) \rightarrow t$ be the victim model which can be a VLM or a Classification model (any models that generates a description text t in response to an input image), where x_{in} is the input image which may be clean (x_{clean}) or adversarial (x_{adv}), p is the input prompt, and t is the resulting output text. In certain tasks, such as image captioning or text retrieval, the input prompt p may remain empty. Let $\mathcal{I}_\phi(x) \rightarrow z$ be a pretrained image encoder and let $G_\psi(t) \rightarrow x_{\text{gen}}$ denote a pretrained text-conditioned image generation model producing image x_{gen}.

3.1. Threat Model

Our framework is designed to detect adversarial attacks, irrespective of the attacker's level of knowledge. In this scenario, there are two parties:

Attacker. The attacker's goal is to generate an adversarial image $x_{\text{adv}} = x_{\text{clean}} + \delta$ that causes the victim model to produce an incorrect caption or classification. The attack can be **targeted**, where the generated text t matches a predefined adversarial target, or **untargeted**, where the model is simply forced to misinterpret or misdescribe the input image. In both cases, the perturbation δ is constrained within an ℓ-norm bounded adversarial budget to ensure minimal perceptibility while maximizing the likelihood of deception. We make no assumptions about the adversary's level of access to the victim model, they may have full knowledge of its architecture, parameters, and training data, or they may operate in a black-box setting with no such information.

Defender. The defender aims to correctly classify input images as either *clean* or *adversarial* by assessing the consistency between the model's interpretation of the input and a reference image generated from the model's textual output. The detection mechanism does not rely on knowledge of the specific adversarial attack strategy and assumes only black-box access to the victim model. Furthermore, the defender does not have access to any ground-truth clean reference image, only the input image x_{in}, which may be either clean or adversarial.

3.2. `MirrorCheck` Pipeline

3.2.1. `Vanilla MirrorCheck`.

The framework is illustrated in Fig. 1. The key observation lies in the deviation of text generated by adversarial images from the content of the input image, which is the primary objective of the attack.

Given an input image x_{in}, we first obtain a textual or a class description using the victim model:

$$t = \mathcal{F}_\theta(x_{\text{in}}, p). \tag{1}$$

This text is then used as input to a pretrained text-to-image model G_ψ, which generates a reconstructed image:

$$x_{\text{gen}} = G_\psi(t). \tag{2}$$

If the input image is clean, the generated image x_{gen} should preserve semantic consistency with x_{in}. However, if x_{in} has been adversarially altered, the perturbation may distort the semantic information, leading to a discrepancy between x_{in} and x_{gen}. To quantify this discrepancy, we compare their feature embeddings obtained as follows:

$$z_{in} = \mathcal{I}_\phi(x_{\text{in}}), \quad z_{\text{gen}} = \mathcal{I}_\phi(x_{\text{gen}}). \tag{3}$$

Subsequently, we employ an adversarial detector $\mathcal{D}(x) \rightarrow [0, 1]$, which categorizes the image into either the "adversarial" class (1) or the "clean" class (0) based on the similarity between the embeddings, with τ serving as the decision threshold parameter, i.e.

$$\mathcal{D}(x) = \begin{cases} 1, & \text{if } S(z_{in}, z_{\text{gen}}) < \tau, \\ 0, & \text{otherwise} \end{cases}.$$

Where S is the similarity metric between these embeddings, we employ the cosine similarity:

$$S(z_{in}, z_{\text{gen}}) = \frac{z_{in} \cdot z_{\text{gen}}}{\|z_{in}\| \|z_{\text{gen}}\|}. \tag{4}$$

The optimal value of τ is determined using the Receiver Operating Characteristic (ROC) curve analysis. Specifically, we identify the point on the ROC curve where the difference between the true positive rate TPR (the proportion of actual adversarial images correctly identified) and the false positive rate FPR (the proportion of clean images incorrectly flagged as adversarial) is maximized. This approach ensures a balanced trade-off between detection sensitivity and robustness, making τ an effective decision threshold for identifying adversarial samples. However, the choice of τ may vary based on the characteristics of the specific text-to-image models or pretrained image encoders used, and we recommend calibrating τ accordingly to account for variations in model behavior.

Intuition behind image-image similarity. Instead of directly comparing x_{in} (the input image) with the generated caption t, we opted to calculate the similarity between x_{in} and x_{gen} (the newly generated image). This decision is based on evidence in the literature indicating that these models struggle with positional relationships and variations in verb usage within sentences. This suggests that VLMs may function more like bags-of-words and, consequently, which could limit their reliability for optimizing cross-modality similarity [38]. Furthermore, we selected this embedding-based similarity metric over conventional metrics like SSIM or FID because those methods may fail to capture semantic equivalence in cases where the T2I model generates a visually different image that is still semantically similar. By utilizing vector embeddings, we aim to maintain high similarity scores in such scenarios, ensuring robustness and reliability even when T2I outputs exhibit variability in their visual representation. Recognizing the potential issue introduced by a single image encoder used for similarity assessment (i.e., if it was used to generate the adversarial samples), the defender employs an ensemble of pretrained image encoders. The final similarity score is obtained by averaging across n predetermined encoders:

$$S_{\text{ensemble}} = \frac{1}{n} \sum_{k=1}^{n} S(z_{in_k}, z_{\text{gen}_k}). \tag{5}$$

3.2.2. `Stochastic MirrorCheck`.

We extend `Vanilla MirrorCheck` through a comprehensive stochastic defense paradigm. While `Vanilla MirrorCheck` employs fixed T2I models and predetermined image encoders (single or ensemble), `Stochastic MirrorCheck` introduces three key innovations that enhance robustness against adaptive attacks: Randomized T2I model selection, Stochastic encoder deployment, and One-Time-Use (OTU) perturbations. This randomization exponentially increases the computational complexity of mounting successful adaptive attacks. It requires adversaries to simultaneously predict **which specific models will be deployed** and **precisely how their parameters will be perturbed**. Formally, the defender maintains a set of M pretrained text-conditioned-image generation models:

$$G = \{G_{\psi_1}, G_{\psi_2}, ..., G_{\psi_M}\}. \tag{6}$$

Table 1. **Similarity scores between original and regenerated images using `Stochastic MirrorCheck`**. The tasks used are image captioning (IC), image description (ID), visual question answering (VQA), and image classification (CL). Clean images consistently achieve high similarity scores, while adversarial examples show degraded similarity, enabling effective detection across models and attack types. Results shown for random CLIP encoder selection with One-Time-Use perturbations across different ensemble sizes and noise scales.

Victim Model	Task	Attack Setting	CLIP Image Encoders (Random Selection + Noise)														
			1 Encoder			3 Encoders			5 Encoders			7 Encoders			10 Encoders		
			5e-6	5e-4	1e-3	5e-6	5e-4	1e-3	5e-6	5e-4	1e-3	5e-6	5e-4	1e-3	5e-6	5e-4	1e-3
UniDiffuser	IC	Clean	0.721	0.624	0.740	0.651	0.701	0.685	0.694	0.647	0.715	0.648	0.693	0.670	0.665	0.670	0.692
		AttackVLM-T	0.502	0.341	0.568	0.370	0.501	0.477	0.494	0.399	0.549	0.402	0.472	0.458	0.424	0.441	0.494
		AttackVLM-Q	0.498	0.294	0.542	0.336	0.448	0.395	0.424	0.332	0.446	0.340	0.411	0.375	0.366	0.372	0.408
BLIP	IC	Clean	0.707	0.610	0.730	0.633	0.686	0.672	0.676	0.628	0.700	0.629	0.675	0.652	0.647	0.653	0.676
		AttackVLM-T	0.481	0.323	0.547	0.349	0.450	0.454	0.419	0.362	0.480	0.353	0.423	0.412	0.375	0.391	0.448
		AttackVLM-Q	0.508	0.299	0.555	0.350	0.460	0.460	0.436	0.346	0.407	0.354	0.424	0.389	0.379	0.384	0.421
BLIP-2	IC	Clean	0.729	0.636	0.744	0.655	0.705	0.687	0.697	0.664	0.718	0.651	0.695	0.684	0.668	0.675	0.695
		AttackVLM-T	0.504	0.345	0.563	0.376	0.473	0.475	0.443	0.381	0.503	0.377	0.447	0.434	0.399	0.413	0.467
		AttackVLM-Q	0.380	0.323	0.484	0.343	0.352	0.408	0.382	0.345	0.401	0.340	0.388	0.395	0.372	0.388	0.421
	ID	Attack-Bard	0.484	0.422	0.536	0.379	0.444	0.498	0.399	0.416	0.468	0.377	0.451	0.427	0.399	0.420	0.461
Img2Prompt	VQA	Clean	0.675	0.563	0.705	0.589	0.652	0.637	0.637	0.585	0.677	0.586	0.638	0.616	0.605	0.613	0.642
		AttackVLM-T	0.482	0.317	0.547	0.345	0.449	0.455	0.416	0.359	0.479	0.349	0.422	0.412	0.372	0.388	0.477
		AttackVLM-Q	0.517	0.309	0.561	0.361	0.470	0.467	0.447	0.356	0.414	0.365	0.431	0.396	0.390	0.392	0.427
LLaVA	VQA	Clean	0.680	0.823	0.755	0.733	0.714	0.741	0.728	0.810	0.748	0.725	0.706	0.733	0.712	0.798	0.742
		Attack-MMFM	0.539	0.724	0.626	0.599	0.596	0.617	0.618	0.710	0.641	0.608	0.602	0.625	0.595	0.695	0.632
OpenFlamingo	VQA	Clean	0.690	0.817	0.756	0.728	0.723	0.743	0.734	0.804	0.749	0.720	0.715	0.735	0.708	0.791	0.742
		Attack-MMFM	0.535	0.714	0.618	0.584	0.609	0.612	0.609	0.701	0.635	0.596	0.614	0.620	0.582	0.688	0.625
MiniGPT-4	VQA	Clean	0.651	0.536	0.684	0.561	0.628	0.618	0.612	0.560	0.646	0.559	0.613	0.593	0.578	0.587	0.620
		AttackVLM-T	0.568	0.457	0.620	0.472	0.548	0.551	0.523	0.481	0.576	0.469	0.532	0.519	0.489	0.504	0.549
DenseNet	CL	Clean	0.543	0.740	0.705	0.671	0.674	0.667	0.692	0.658	0.695	0.665	0.688	0.671	0.652	0.660	0.679
		FGSM	0.444	0.666	0.572	0.537	0.548	0.553	0.579	0.521	0.584	0.535	0.572	0.558	0.518	0.541	0.567
		BIM	0.507	0.713	0.593	0.554	0.532	0.579	0.601	0.542	0.598	0.548	0.586	0.571	0.531	0.553	0.581
		PGD	0.495	0.705	0.585	0.546	0.524	0.571	0.593	0.534	0.590	0.540	0.578	0.563	0.523	0.545	0.573
		DeepFool	0.475	0.690	0.565	0.525	0.510	0.555	0.575	0.515	0.570	0.520	0.560	0.545	0.505	0.525	0.555
		C&W	0.460	0.680	0.555	0.515	0.500	0.545	0.565	0.505	0.560	0.510	0.550	0.535	0.495	0.515	0.545
MobileNet	CL	Clean	0.668	0.790	0.729	0.704	0.705	0.719	0.745	0.698	0.738	0.702	0.726	0.712	0.688	0.695	0.721
		FGSM	0.520	0.712	0.606	0.612	0.585	0.607	0.635	0.598	0.629	0.605	0.618	0.610	0.585	0.592	0.615
		BIM	0.503	0.693	0.581	0.565	0.538	0.576	0.605	0.572	0.598	0.575	0.590	0.582	0.558	0.565	0.585
		PGD	0.495	0.685	0.573	0.557	0.530	0.568	0.597	0.564	0.590	0.567	0.582	0.574	0.550	0.557	0.577
		DeepFool	0.475	0.670	0.555	0.540	0.515	0.552	0.580	0.548	0.573	0.550	0.565	0.557	0.535	0.542	0.562
		C&W	0.465	0.660	0.545	0.530	0.505	0.542	0.570	0.538	0.563	0.540	0.555	0.547	0.525	0.532	0.552

where x_{gen} is generated by a one randomly selected generation model. Additionally, the defender maintains a model zoo of N pretrained image encoders. At detection time, a subset of n encoders is randomly selected:

$$\mathcal{I}_{\text{subset}} = \{\mathcal{I}_{\phi_1}, \mathcal{I}_{\phi_2}, ..., \mathcal{I}_{\phi_n}\}, \quad n \ll N. \tag{7}$$

To further enhance robustness, each selected encoder undergoes `OTU` perturbation, where its parameters are modified by injecting small random noise γ:

$$\hat{\phi}_k = \phi_k + \gamma, \quad \forall k \in \mathcal{I}_{\text{subset}}. \tag{8}$$

This ensures that the attacker cannot reliably optimize against a fixed set of encoders. The perturbed encoders extract feature embeddings from both the original and regenerated images:

$$z_{i_k} = \mathcal{I}_{\hat{\phi}_{i_k}}(x_{\text{in}}), \quad z_{\text{gen},i_k} = \mathcal{I}_{\hat{\phi}_{i_k}}(x_{\text{gen}}). \tag{9}$$

Finally, S_{ensemble} is computed.

3.3. Robustness to Adaptive Attacks

Following best practices established by Athalye et al. [15], we evaluate `MirrorCheck` under strong white-box adaptive attacks to ensure our defense does not rely on gradient obfuscation or hidden randomness. We design a worst-case attacker with full knowledge of the victim model $\mathcal{F}_\theta$, the entire model zoo of image encoders $\{\mathcal{I}_{\phi_j}\}_{j=1}^N$, the text-to-image (T2I) models $\{G_\psi\}$, and the detection mechanism.

The attacker's goal is twofold: (i) to induce the victim model to produce a target text or class, and (ii) to evade detection by maximizing the similarity between the adversarial image and its regenerated counterpart. A practical challenge is that the victim's VLM output is inherently discrete; it produces sampled tokens, breaking the gradient path. To resolve this, we extend the classic Backward Pass Differentiable Approximation (BPDA) [15] method: the attacker uses the **true** discrete text in the forward pass to condition the T2I model G_ψ, but replaces the non-differentiable text sampling with a learnable Adapter $\mathcal{A}$ in the backward pass.

Table 2. **Detection accuracy of `Stochastic MirrorCheck` across diverse victim models and attack types.** The method achieves consistently high detection rates (65-99%) across VLM attacks (AttackVLM, Attack-Bard, Attack-MMFM) and classification attacks (FGSM, BIM, PGD, DeepFool, C&W), demonstrating robust performance with randomized encoder selection and One-Time-Use perturbations.

Victim Model	Setting	CLIP Image Encoders (Random Selection + Noise)														
		1 Encoder			3 Encoders			5 Encoders			7 Encoders			10 Encoders		
		5e-6	5e-4	1e-3	5e-6	5e-4	1e-3	5e-6	5e-4	1e-3	5e-6	5e-4	1e-3	5e-6	5e-4	1e-3
UniDiffuser	AttackVLM-T	0.913	0.895	0.903	0.925	0.943	0.858	0.933	0.890	0.910	0.908	0.912	0.915	0.920	0.918	0.917
	AttackVLM-Q	0.955	0.902	0.968	0.952	0.968	0.937	0.973	0.975	0.980	0.977	0.978	0.980	0.975	0.973	0.992
BLIP	AttackVLM-T	0.905	0.908	0.908	0.918	0.932	0.903	0.922	0.915	0.925	0.930	0.930	0.918	0.930	0.923	0.925
	AttackVLM-Q	0.928	0.887	0.943	0.913	0.937	0.915	0.958	0.943	0.963	0.962	0.947	0.955	0.965	0.945	0.955
BLIP-2	AttackVLM-T	0.912	0.892	0.912	0.920	0.932	0.907	0.923	0.923	0.928	0.927	0.935	0.932	0.930	0.927	0.938
	AttackVLM-Q	0.945	0.903	0.967	0.948	0.957	0.932	0.975	0.97	0.985	0.982	0.978	0.977	0.978	0.970	0.990
	Attack-Bard	0.883	0.790	0.827	0.890	0.873	0.900	0.903	0.952	0.903	0.942	0.913	0.902	0.927	0.920	0.938
Img2Prompt	AttackVLM-T	0.848	0.840	0.843	0.878	0.873	0.853	0.882	0.860	0.867	0.878	0.875	0.883	0.895	0.878	0.882
	AttackVLM-Q	0.880	0.806	0.907	0.861	0.863	0.855	0.886	0.857	0.905	0.870	0.877	0.905	0.887	0.882	0.920
LLaVA	Attack-MMFM	0.788	0.728	0.733	0.812	0.762	0.738	0.818	0.812	0.783	0.832	0.815	0.812	0.845	0.833	0.820
OpenFlamingo	Attack-MMFM	0.800	0.740	0.765	0.777	0.750	0.750	0.807	0.785	0.767	0.800	0.780	0.797	0.797	0.793	0.785
MiniGPT-4	AttackVLM-T	0.642	0.623	0.632	0.660	0.660	0.642	0.655	0.665	0.667	0.655	0.665	0.657	0.655	0.665	0.655
DenseNet	FGSM	0.850	0.840	0.800	0.845	0.835	0.795	0.852	0.842	0.802	0.847	0.837	0.798	0.849	0.839	0.801
	BIM	0.860	0.830	0.830	0.855	0.825	0.825	0.862	0.832	0.832	0.857	0.827	0.828	0.859	0.829	0.831
	PGD	0.845	0.825	0.815	0.840	0.820	0.810	0.847	0.827	0.817	0.842	0.822	0.812	0.844	0.824	0.814
	DeepFool	0.835	0.815	0.795	0.830	0.810	0.790	0.837	0.817	0.797	0.832	0.812	0.792	0.834	0.814	0.794
	C&W	0.825	0.805	0.785	0.820	0.800	0.780	0.827	0.807	0.787	0.822	0.802	0.782	0.824	0.804	0.784
MobileNet	FGSM (0.3)	0.780	0.770	0.750	0.775	0.765	0.745	0.782	0.772	0.752	0.777	0.767	0.747	0.779	0.769	0.749
	FGSM (0.1)	0.850	0.850	0.790	0.845	0.845	0.785	0.852	0.852	0.792	0.847	0.847	0.787	0.849	0.849	0.789
	BIM	0.790	0.790	0.780	0.785	0.785	0.775	0.792	0.792	0.782	0.787	0.787	0.777	0.789	0.789	0.779
	PGD	0.775	0.775	0.765	0.770	0.770	0.760	0.777	0.777	0.767	0.772	0.772	0.762	0.774	0.774	0.764
	DeepFool	0.760	0.750	0.740	0.755	0.745	0.735	0.762	0.752	0.742	0.757	0.747	0.737	0.759	0.749	0.739
	C&W	0.745	0.735	0.725	0.740	0.730	0.720	0.747	0.737	0.727	0.742	0.732	0.722	0.744	0.734	0.724

This Adapter maps the VLM's continuous hidden state to the T2I conditioning space, providing a differentiable surrogate path for gradients. Additionally, we apply Expectation over Transformation (EoT) [23] to handle all sources of randomness in the pipeline, averaging over the stochastic sampling noise η in G_ψ, and the one-time-use (OTU) perturbations γ applied to encoder weights. Because the attacker does not know which encoders will be sampled, they must optimize over the entire model zoo. The resulting adaptive attack objective is:

$$\min_{\delta:\|\delta\|_\infty \le \varepsilon} \quad L_{\mathcal{F}_\theta}(x+\delta) + \lambda \cdot \mathcal{L}_{\text{det}}, \tag{10}$$

where the detection loss $\mathcal{L}_{\text{det}}$ is defined as:

$$\mathcal{L}_{\text{det}} = \mathbb{E}_{\psi,\,\eta,\,\gamma}\Big[1 - \frac{1}{N}\sum_{j=1}^{N} S(\mathbf{z}_1, \mathbf{z}_2)\Big], \tag{11}$$
$$\mathbf{z}_1 = \mathcal{I}_{\phi_j+\gamma}(x+\delta),$$
$$\mathbf{z}_2 = \mathcal{I}_{\phi_j+\gamma}(G_\psi(\mathcal{F}_\theta(x+\delta);\,\eta)).$$

In this formulation, $L_{\mathcal{F}_\theta}(x+\delta)$ denotes the attacker's primary task objective that forces the victim VLM to output a desired target caption or class. For example, in the original attack setting [7], this is defined by aligning the adversarial input with a target image generated from a surrogate image encoder $\mathcal{E}$:

$$\arg\min_{\delta:\|\delta\|_\infty \le \varepsilon} d\big(\mathcal{E}(x+\delta),\, \mathcal{E}(x_{\text{ref}})\big),$$

where x_{ref} is the target image.The attacker minimizes the expected detection score over all relevant randomness to robustly evade detection in the worst case. To train the Adapter $\mathcal{A}$, we construct a dataset of 34,000 paired feature representations from clean ImageNet images.

Image-to-text (VLM) features. For each image, we use UniDiffuser as a captioning pipeline to generate a caption and extract the corresponding intermediate captioning representation by sampling from a trained DPM solver and encoding it via the model's decoder head, yielding $\mathbf{z}_{\text{VLM}} \in \mathbb{R}^{77\times 64}$.

Text-to-image (T2I) features. The generated caption is then passed to a Stable Diffusion pipeline, from which we extract the internal prompt embedding $\mathbf{z}_{\text{T2I}} \in \mathbb{R}^{77\times 768}$, using deterministic settings (1 denoising step). The Adapter $\mathcal{A}$ is implemented as a lightweight convolutional MLP: three convolutional layers process the input $\mathbf{z}_{\text{VLM}}$ as a 2D tensor of shape $1 \times 77 \times 64$, with ReLU activations and max-pooling. The resulting feature map is flattened and passed

through four fully connected layers to produce a final output of shape 77×768, matching the T2I embedding dimensionality. We optimize the Adapter using an L2 regression loss:

$$\mathcal{L}_{\text{Adapter}} = \|\mathcal{A}(\mathbf{z}_{\text{VLM}}) - \mathbf{z}_{\text{T2I}}\|_2^2 .$$

4. Experiments

We evaluate `MirrorCheck` variants across three key dimensions: (1) performance in unimodal and multimodal tasks, (2) comparison against baselines, and (3) robustness against adaptive attacks. All experiments are run three times on 2000 images (1000 clean and 1000 attacked) and use open-source models for reproducibility.

4.1. Implementation Details

Victim Models. We evaluate on diverse architectures: *Multimodal models* including UniDiffuser [39], BLIP [2], Img2Prompt [40], BLIP-2 [3], LLaVA [41], OpenFlamingo [42], and MiniGPT-4 [4]; and *Unimodal models* including DenseNet [43] and MobileNet [44] for classification tasks.

Adversarial Attack Settings. We evaluate against various attack strategies using settings from their original papers: *Targeted attacks* such as AttackVLM transfer and query-based variants [7] on ImageNet-1K validation images [45] with randomly selected MS-COCO captions [46] as targets; *Untargeted attacks* including Attack-Bard [24] on NIPS17 dataset [47], Attack-MMFM [48] on COCO 2014 captioning tasks [46], and standard attacks (FGSM [20], BIM [49], PGD [50], DeepFool [51], C&W) on CIFAR-10 [52] and ImageNet [45]. All attack parameters follow the original implementations.

T2I Models. Our T2I model zoo includes Stable Diffusion v1.4/v1.5 [53], UniDiffuser [39], and ControlNet [54]. All generation uses 50 timesteps, producing 512×512 pixel outputs. For `Stochastic MirrorCheck`, one T2I model is randomly selected per inference.

Image Encoders. Our encoder collection includes OpenAI CLIP [55] variants, OpenCLIP [56] models, and pretrained VGG16 [57] and ResNet-50 [58] from PyTorch. Model-specific preprocessing is applied to both input and generated images. For `Stochastic MirrorCheck`, $n \in \{1, 3, 5, 7, 10\}$ encoders are randomly selected per inference.

MirrorCheck Config. `Vanilla MirrorCheck` employs fixed T2I models (primarily SD v1.4) and predetermined encoder sets. `Stochastic MirrorCheck` introduces randomization across three dimensions: (1) random T2I model selection from our zoo, (2) random encoder selection (single or ensemble), and (3) One-Time-Use (OTU) perturbations with Gaussian noise scales $\{5e^{-6}, 5e^{-4}, 1e^{-3}\}$ applied to encoder parameters before each inference.

Baseline. We evaluate against both classical detection approaches adapted from image classification and recent VLM-specific defense methods. (**Unimodal Baselines.**) **MagNet** [30] uses autoencoders to detect adversarial inputs via reconstruction error analysis on the manifold of clean images. **PuVAE** [59] employs variational autoencoders for adversarial purification through latent space projection. **FeatureSqueeze** [29] creates "squeezed" input versions and compares model predictions for discrepancy detection. **DiffPure** [11] leverages diffusion models for adversarial purification. For VLM adaptation, we train MagNet and PuVAE autoencoders on ImageNet to learn clean image manifolds, while FeatureSqueeze compares captions generated from original versus squeezed inputs. (**Multimodal Baselines.**) **CIDER** [60] detects adversarial examples in Multimodal Large Language Models by measuring the discrepancy in cross-modal semantic alignment between text-image pairs before and after diffusion-based denoising. **Naive** implements an ensemble voting approach where the same input is passed to five VLMs (including the victim model), flagging inputs with significant response divergence as adversarial. **CLIP** calculates similarity scores between input images and victim model responses using CLIP embeddings [55] for detection. **JailGuard** [61], **SmoothVLM** [37], and **DPS** [36] are mutation-based approaches defending against jailbroken prompts. We adapt JailGuard and SmoothVLM by mutating inputs using filters and patch-based perturbations, while DPS leverages cropping techniques. All methods flag inputs based on response divergence after mutation.

4.2. Benchmarking `MirrorCheck`'s Performance

Table 1 shows similarity scores using `Stochastic MirrorCheck` across diverse victim models and attack scenarios with randomized encoder selection and `OTU` perturbations. Clean images consistently achieve high similarity scores, demonstrating that legitimate inputs maintain strong semantic consistency when processed through our stochastic detection framework. In contrast, adversarial examples exhibit lower similarity scores. This gap between clean and adversarial similarity scores enables effective detection. The results demonstrate robust performance across various ensemble sizes (1, 3, 5, 7, 10 encoders) and noise perturbation scales (5e-6, 5e-4, 1e-3). Notably, larger ensemble sizes generally provide more stable detection boundaries, while different noise scales offer varying levels of adaptive attack resistance. Leveraging the observed similarity scores, we compute the *detection accuracy* of our method as the ratio of correctly identified clean and adversarial images to the total number of images for each attack. As shown in Table 2, the method maintains consistent discriminative capabilities and detection accuracies (as high as 99%) across unimodal and multimodal ar-

Table 3. **Adversarial detection performance across VLM attacks.** Our MirrorCheck variants consistently outperform both unimodal approaches and multimodal VLM-specific methods, achieving superior detection rates as high as 0.99. Best results in **bold**.

Victim Model	Attack Setting	Unimodal Approaches				Multimodal Approaches						Ours	
		FS	MagNet	PuVAE	DiffPure	CIDER	Naive	CLIP	JailGuard	SmoothVLM	DPS	MC	Stochastic-MC
UniDiffuser	AttackVLM-T	0.56	0.74	0.51	0.80	0.84	0.68	0.59	0.81	0.82	0.83	**0.96**	0.95
	AttackVLM-Q	0.65	0.85	0.70	0.81	0.80	0.65	0.57	0.83	0.83	0.85	**0.98**	**0.98**
BLIP	AttackVLM-T	0.52	0.60	0.50	0.71	0.81	0.66	0.61	0.79	0.77	0.81	0.90	**0.93**
	AttackVLM-Q	0.57	0.65	0.80	0.76	0.85	0.64	0.55	0.84	0.81	0.84	0.89	**0.97**
BLIP-2	AttackVLM-T	0.61	0.73	0.52	0.80	0.84	0.70	0.62	0.82	0.80	0.86	0.93	**0.94**
	AttackVLM-Q	0.61	0.85	0.72	0.83	0.77	0.67	0.58	0.80	0.78	0.83	0.92	**0.99**
	Attack-Bard	-	-	-	0.79	0.87	0.65	0.58	0.89	0.87	0.95	**0.98**	0.95
Img2Prompt	AttackVLM-T	0.51	0.56	0.50	0.67	0.83	0.61	0.56	0.83	0.83	0.86	0.79	**0.90**
	AttackVLM-Q	-	0.65	0.78	0.69	0.79	0.60	0.55	0.81	0.74	0.82	0.85	**0.92**
LLaVA	Attack-MMFM	-	-	-	0.67	0.83	0.62	0.52	**0.85**	**0.85**	**0.85**	0.82	**0.85**
OpenFlamingo	Attack-MMFM	-	-	-	0.65	0.84	0.60	0.51	**0.87**	0.84	0.86	0.81	0.81
MiniGPT-4	AttackVLM-T	0.54	0.51	0.53	0.62	**0.85**	0.57	0.51	**0.85**	0.80	**0.85**	0.66	0.67

chitectures under diverse attack settings and across different downstream tasks. We present complete similarity scores and detection results using different encoders and T2I models for our vanilla variant in Appendix C.

4.3. Comparison with Baselines

Table 3 presents adversarial detection performance across a diverse set of vision-language models (VLMs) and attack scenarios. We compare our proposed MirrorCheck variants (**MC** and **Stochastic-MC**) against both unimodal defenses (FS, MagNet, PuVAE, DiffPure) and multimodal methods specifically tailored for VLMs (CIDER, Naive, CLIP, JailGuard, SmoothVLM, DPS). Overall, MirrorCheck consistently achieves superior detection rates, often by a large margin. For example, on UniDiffuser under the AttackVLM-Q setting, Stochastic-MC attains a detection score of **0.98**, outperforming the next best baseline by more than 0.13. Similarly, on BLIP-2 with AttackVLM-Q, MirrorCheck reaches **0.99**. Our approach demonstrates robustness across both text- and query-based attacks, as well as multimodal fusion attacks (e.g., Attack-MMFM), where existing unimodal defenses fail to generalize. Notably, even against strong baselines like DiffPure and JailGuard, which are widely used in current multimodal adversarial defenses, MirrorCheck maintains consistent gains without requiring model retraining or access to internal gradients/logits. This highlights the practicality and scalability of our method for real-world deployment.

4.4. Impact of Scaling Factor

From Tab. 2, we find that using moderate amounts of noise (scaling factors $\leq 5\times10^{-4}$) works best for detecting attacks across all models and attack types. This happens because larger scaling factors disrupts the model's learned patterns, making it harder to distinguish between normal and malicious inputs. The same pattern holds for different types of attacks, though the effect is less noticeable for simpler classification attacks.

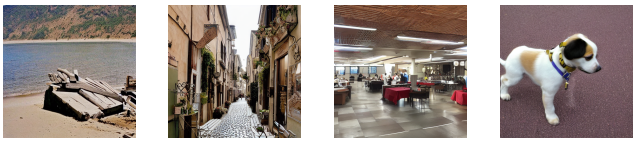

Figure 2. Images generated from Stable Diffusion conditioned on Adapter outputs computed from UniDiffuser's captioning features.

4.5. Impact of Encoder Size

We also observe that using more encoders generally improves attack detection rates. However, the benefits start to plateau after 5–7 encoders (this is not the case for adaptive attacks, see Sec. 4.6). This suggests that while having multiple encoders helps by providing diverse perspectives, adding too many encoders provides minimal improvement while increasing computational costs.

4.6. Adaptive Attack

The Adapter $\mathcal{A}$ is trained using the Adam optimizer over 500 epochs with a batch size of 64. To qualitatively verify the effectiveness of the trained Adapter, we generate images using Stable Diffusion by conditioning directly on the Adapter output $\mathcal{A}(\mathbf{z}_{\text{VLM}})$, where $\mathbf{z}_{\text{VLM}}$ is the feature produced by UniDiffuser's image-to-text captioning pipeline using ImageNet test images. As shown in Fig. 2, the resulting images are visually coherent and reflect the semantics of the original input images, demonstrating that the Adapter successfully bridges the representation gap between the captioning and generation pipelines.

Detection Accuracy Under Adaptive Attacks. We evaluate our method under a white-box adaptive attack follow-

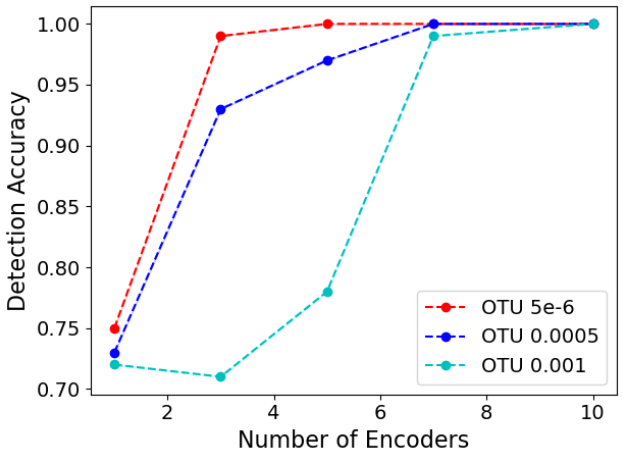

Figure 3. Average detection accuracy over three runs with different random seeds as a function of the number of encoders for different noise levels under adaptive attack.

ing the BPDA+EoT strategy described in §3.3. We randomly sample 10 image–target pairs from ImageNet and optimize an ℓ_∞-bounded perturbation ($\varepsilon = 8$) over 100 PGD steps with a step size of 1. The attacker has full access to the victim model, the trained Adapter, and the entire detection pipeline. The attack is implemented using Stable Diffusion (v1.5) for generation, UniDiffuser for captioning, and the model zoo of 10 CLIP and OpenCLIP encoders. We apply Stochastic MirrorCheck for detection. Figure 3 illustrates the accuracy trends. We observe that detection accuracy consistently improves with the number of encoders in the ensemble. With just 3 encoders, the accuracy already approaches 95% for low and medium OTU noise (5e-6, 5e-4), and with 7 or more encoders, all configurations exceed 98%. At the highest noise level (0.001), the improvement is more gradual, with accuracy increasing from 71% (single encoder) to 99% (10 encoders), highlighting the compounding benefits of both encoder diversity and stochasticity. These results confirm that even under strong adaptive attacks, increasing ensemble size and applying per-run perturbation noise significantly enhances detection robustness.

4.7. Additional ablation and Insights

We perform additional ablation studies and analyses to further explain why `MirrorCheck` is effective (Appendices B and C). We reinterpret our framework from an autoencoder perspective (Appendix B.1) and highlight key intuitive observations (Appendix B.2) that reveal its strengths and potential failure cases. Our results show that `MirrorCheck` is model-agnostic and generalizes across diverse image encoders and T2I models. We also introduce an adaptive attack with a different objective (Appendix C) and show that `MirrorCheck` remains robust. Finally, we study the impact of the clean-to-adversarial ratio and provide qualitative visualizations showing `MirrorCheck` in action.

4.8. Computational Efficiency

Our experiments were carried out on a NVIDIA Quadro RTX A6000 48GB GPU. The entire defense pipeline takes approximately 15 seconds per image. Within this process, obtaining a caption from the victim VLM model takes around 0.2 seconds, generating an image takes about 5 seconds, and calculating similarity requires approximately 10 seconds. However, this is the worst case scenario and there are multiple methods to improve this time i.e., reducing timesteps for generation from 50 to 10 allows the pipeline process an image in just 1.2 seconds with a little compromise in detection performance.

5. Conclusion

We introduce `MirrorCheck`, a novel adversarial detection framework that leverages T2I generation and similarity analysis. Our `Stochastic MirrorCheck` variant employs randomized model selection and One-Time-Use perturbations to create robust defenses against adaptive attacks. Comprehensive evaluation across diverse VLM architectures and attack scenarios demonstrates superior performance compared to adapted baseline methods, with detection accuracies going as high as 99%. This work establishes a new paradigm for multimodal adversarial defense and provides a foundation for securing next-generation AI systems against evolving threats.

References

[1] F. Bao, S. Nie, K. Xue, Y. Cao, C. Li, H. Su, and J. Zhu, "All are worth words: A vit backbone for diffusion models," in *CVPR*, 2023. 1, 4

[2] J. Li, D. Li, C. Xiong, and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," 2022. 6, 1, 4, 5

[3] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *International conference on machine learning*, pp. 19730–19742, PMLR, 2023. 6, 1, 4, 5

[4] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny, "Minigpt-4: Enhancing vision-language understanding with advanced large language models," *arXiv preprint arXiv:2304.10592*, 2023. 6, 4

[5] D. Li, J. Li, H. Le, G. Wang, S. Savarese, and S. C. Hoi, "LAVIS: A one-stop library for language-vision intelligence," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, (Toronto, Canada), pp. 31–41, Association for Computational Linguistics, July 2023. 1

[6] Z. Yin, M. Ye, T. Zhang, T. Du, J. Zhu, H. Liu, J. Chen, T. Wang, and F. Ma, "Vlattack: Multimodal adversarial

attacks on vision-language tasks via pre-trained models," *arXiv preprint arXiv:2310.04655*, 2023. 1, 2

[7] Y. Zhao, T. Pang, C. Du, X. Yang, C. Li, N.-M. Cheung, and M. Lin, "On evaluating adversarial robustness of large vision-language models," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 2, 5, 6

[8] S. Vemprala, R. Bonatti, A. Bucker, and A. Kapoor, "Chatgpt for robotics: Design principles and model abilities," 2023. 1

[9] J. H. Metzen, T. Genewein, V. Fischer, and B. Bischoff, "On detecting adversarial perturbations," *arXiv preprint arXiv:1702.04267*, 2017. 1

[10] K. Roth, Y. Kilcher, and T. Hofmann, "The odds are odd: A statistical test for detecting adversarial examples," in *International Conference on Machine Learning*, pp. 5498–5507, PMLR, 2019. 1, 2

[11] W. Nie, B. Guo, Y. Huang, C. Xiao, A. Vahdat, and A. Anandkumar, "Diffusion models for adversarial purification," in *International Conference on Machine Learning (ICML)*, 2022. 1, 2, 6

[12] P. Samangouei, M. Kabkab, and R. Chellappa, "Defense-GAN: Protecting classifiers against adversarial attacks using generative models," in *International Conference on Learning Representations*, 2018. 1, 2

[13] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deepd learning models resistant to adversarial attacks," in *International Conference on Learning Representations*, 2018. 1, 2

[14] J. Cohen, E. Rosenfeld, and Z. Kolter, "Certified adversarial robustness via randomized smoothing," in *Proceedings of the 36th International Conference on Machine Learning* (K. Chaudhuri and R. Salakhutdinov, eds.), vol. 97 of *Proceedings of Machine Learning Research*, pp. 1310–1320, PMLR, 09–15 Jun 2019. 1, 2

[15] A. Athalye, N. Carlini, and D. A. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *International Conference on Machine Learning*, 2018. 1, 2, 4

[16] S. Wang, J. Zhang, Z. Yuan, and S. Shan, "Pre-trained model guided fine-tuning for zero-shot adversarial robustness," 2024. 1

[17] C. Mao, S. Geng, J. Yang, X. Wang, and C. Vondrick, "Understanding zero-shot adversarial robustness for large-scale models," 2023.

[18] Y. Xie, M. Fang, R. Pi, and N. Gong, "Gradsafe: Detecting jailbreak prompts for llms via safety-critical gradient analysis," *arXiv preprint arXiv:2402.13494*, 2024. 2

[19] Y. Zhang, R. Xie, J. Chen, X. Sun, and Y. Wang, "Pip: Detecting adversarial examples in large vision-language models via attention patterns of irrelevant probe questions," in *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 11175–11183, 2024. 1, 2

[20] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2015. 2, 6, 1

[21] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," *arXiv preprint arXiv:1611.01236*, 2016.

[22] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*, pp. 39–57, Ieee, 2017. 2, 1, 3

[23] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok, "Synthesizing robust adversarial examples," in *International conference on machine learning*, pp. 284–293, PMLR, 2018. 2, 5

[24] Y. Dong, H. Chen, J. Chen, Z. Fang, X. Yang, Y. Zhang, Y. Tian, H. Su, and J. Zhu, "How robust is google's bard to adversarial image attacks?," 2023. 2, 6, 1

[25] Y. Cao, D. Li, M. Fang, T. Zhou, J. Gao, Y. Zhan, and D. Tao, "Tasa: Deceiving question answering models by twin answer sentences attack," *arXiv preprint arXiv:2210.15221*, 2022. 2

[26] V. Kovatchev, T. Chatterjee, V. S. Govindarajan, J. Chen, E. Choi, G. Chronis, A. Das, K. Erk, M. Lease, J. J. Li, *et al.*, "longhorns at dadc 2022: How many linguists does it take to fool a question answering model? a systematic approach to adversarial attacks," *arXiv preprint arXiv:2206.14729*, 2022.

[27] J. Zhang, Q. Yi, and J. Sang, "Towards adversarial attack on vision-language pre-training models," in *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 5005–5013, 2022.

[28] N. Aafaq, N. Akhtar, W. Liu, M. Shah, and A. Mian, "Controlled caption generation for images through adversarial attacks," *arXiv preprint arXiv:2107.03050*, 2021. 2

[29] W. Xu, D. Evans, and Y. Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," *ArXiv*, vol. abs/1704.01155, 2017. 2, 6

[30] D. Meng and H. Chen, "Magnet: a two-pronged defense against adversarial examples," in *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, pp. 135–147, 2017. 2, 6, 4

[31] U. Hwang, J. Park, H. Jang, S. Yoon, and N. I. Cho, "Puvae: A variational autoencoder to purify adversarial examples," *IEEE Access*, vol. 7, pp. 126582–126593, 2019. 2

[32] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," in *International Conference on Learning Representations*, 2017. 2

[33] H. Salman, M. Sun, G. Yang, A. Kapoor, and J. Z. Kolter, "Denoised smoothing: a provable defense for pretrained classifiers," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, (Red Hook, NY, USA), Curran Associates Inc., 2020. 2

[34] F. Tramer, N. Carlini, W. Brendel, and A. Madry, "On adaptive attacks to adversarial example defenses," *Advances in neural information processing systems*, vol. 33, pp. 1633–1645, 2020. 2

[35] N. Carlini and D. Wagner, "Defensive distillation is not robust to adversarial examples," *arXiv preprint arXiv:1607.04311*, 2016. 2

[36] Q. Zhou, T. Li, Q. Guo, D. Wang, Y. Lin, Y. Liu, and J. S. Dong, "Defending lvlms against vision attacks through partial-perception supervision," *arXiv preprint arXiv:2412.12722*, 2024. 2, 6

[37] J. Sun, C. Wang, J. Wang, Y. Zhang, and C. Xiao, "Safeguarding vision-language models against patched visual prompt injectors," *arXiv preprint arXiv:2405.10529*, 2024. 2, 6

[38] M. Yuksekgonul, F. Bianchi, P. Kalluri, D. Jurafsky, and J. Zou, "When and why vision-language models behave like bags-of-words, and what to do about it?," in *TInternational Conference on Learning Representations*, 2022. 3

[39] F. Bao, S. Nie, K. Xue, C. Li, S. Pu, Y. Wang, G. Yue, Y. Cao, H. Su, and J. Zhu, "One transformer fits all distributions in multi-modal diffusion at scale," in *International Conference on Machine Learning*, pp. 1692–1717, PMLR, 2023. 6, 4, 5

[40] J. Guo, J. Li, D. Li, A. M. Huat Tiong, B. Li, D. Tao, and S. Hoi, "From images to textual prompts: Zero-shot visual question answering with frozen large language models," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10867–10877, 2023. 6, 4, 5

[41] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," 2023. 6

[42] A. Awadalla, I. Gao, J. Gardner, J. Hessel, Y. Hanafy, W. Zhu, K. Marathe, Y. Bitton, S. Gadre, S. Sagawa, J. Jitsev, S. Kornblith, P. W. Koh, G. Ilharco, M. Wortsman, and L. Schmidt, "Openflamingo: An open-source framework for training large autoregressive vision-language models," *arXiv preprint arXiv:2308.01390*, 2023. 6

[43] F. Iandola, M. Moskewicz, S. Karayev, R. Girshick, T. Darrell, and K. Keutzer, "Densenet: Implementing efficient convnet descriptor pyramids," *arXiv preprint arXiv:1404.1869*, 2014. 6

[44] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018. 6

[45] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009. 6

[46] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755, Springer, 2014. 6

[47] A. Kurakin, I. Goodfellow, S. Bengio, Y. Dong, F. Liao, M. Liang, T. Pang, J. Zhu, X. Hu, C. Xie, *et al.*, "Adversarial attacks and defences competition," in *The NIPS'17 Competition: Building Intelligent Systems*, pp. 195–231, Springer, 2018. 6

[48] C. Schlarmann and M. Hein, "On the adversarial robustness of multi-modal foundation models," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3677–3685, 2023. 6, 1

[49] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Artificial intelligence safety and security*, pp. 99–112, Chapman and Hall/CRC, 2018. 6

[50] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2019. 6

[51] S. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: A simple and accurate method to fool deep neural networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Los Alamitos, CA, USA), pp. 2574–2582, IEEE Computer Society, jun 2016. 6, 2

[52] A. Krizhevsky, "Learning multiple layers of features from tiny images," 2009. 6

[53] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," 2022. 6

[54] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3836–3847, 2023. 6, 4

[55] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning* (M. Meila and T. Zhang, eds.), vol. 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763, PMLR, 18–24 Jul 2021. 6, 1

[56] G. Ilharco, M. Wortsman, R. Wightman, C. Gordon, N. Carlini, R. Taori, A. Dave, V. Shankar, H. Namkoong, J. Miller, H. Hajishirzi, A. Farhadi, and L. Schmidt, "Openclip," 2021. If you use this software, please cite it as below. 6

[57] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014. 6

[58] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016. 6

[59] Y. Pu, Z. Gan, R. Henao, X. Yuan, C. Li, A. Stevens, and L. Carin, "Variational autoencoder for deep learning of images, labels and captions," *Advances in neural information processing systems*, vol. 29, 2016. 6, 4

[60] Y. Xu, X. Qi, Z. Qin, and W. Wang, "Cross-modality information check for detecting jailbreaking in multimodal large language models," in *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 13715–13726, 2024. 6

[61] X. Zhang, C. Zhang, T. Li, Y. Huang, X. Jia, M. Hu, J. Zhang, Y. Liu, S. Ma, and C. Shen, "Jailguard: A universal detection framework for llm prompt-based attacks," *arXiv preprint arXiv:2312.10766*, 2023. 6

[62] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2023. 1

[63] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021. 1

[64] X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei, *et al.*, "Oscar: Object-semantics aligned pre-training for vision-language tasks," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16*, pp. 121–137, Springer, 2020. 1

[65] J. Li, R. Selvaraju, A. Gotmare, S. Joty, C. Xiong, and S. C. H. Hoi, "Align before fuse: Vision and language representation learning with momentum distillation," *Advances in neural information processing systems*, vol. 34, pp. 9694–9705, 2021. 1

[66] L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang, "Visualbert: A simple and performant baseline for vision and language," *arXiv preprint arXiv:1908.03557*, 2019. 1

[67] W. Su, X. Zhu, Y. Cao, B. Li, L. Lu, F. Wei, and J. Dai, "Vl-bert: Pre-training of generic visual-linguistic representations," *arXiv preprint arXiv:1908.08530*, 2019. 1

[68] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li, and T. Duerig, "Scaling up visual and vision-language representation learning with noisy text supervision," in *International conference on machine learning*, pp. 4904–4916, PMLR, 2021. 1

[69] A. Singh, R. Hu, V. Goswami, G. Couairon, W. Galuba, M. Rohrbach, and D. Kiela, "Flava: A foundational language and vision alignment model," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15617–15629, 2021. 1

[70] H. Bao, W. Wang, L. Dong, Q. Liu, O. K. Mohammed, K. Aggarwal, S. Som, S. Piao, and F. Wei, "VLMo: Unified vision-language pre-training with mixture-of-modality-experts," in *Advances in Neural Information Processing Systems* (A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, eds.), 2022. 1

[71] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican, *et al.*, "Gemini: a family of highly capable multimodal models," *arXiv preprint arXiv:2312.11805*, 2023. 1

[72] H. Chen, Y. Zhang, Y. Dong, X. Yang, H. Su, and J. Zhu, "Rethinking model ensemble in transfer-based adversarial attacks," *arXiv preprint arXiv:2303.09105*, 2023. 1

[73] Y. Liu, X. Chen, C. Liu, and D. Song, "Delving into transferable adversarial examples and black-box attacks," *arXiv preprint arXiv:1611.02770*, 2016. 1

[74] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013. 1

[75] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 427–436, 2015. 1

[76] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *2016 IEEE European symposium on security and privacy (EuroS&P)*, pp. 372–387, IEEE, 2016. 1

[77] R. Feinman, R. R. Curtin, S. Shintre, and A. B. Gardner, "Detecting adversarial samples from artifacts," *arXiv preprint arXiv:1703.00410*, 2017. 2

[78] Y. Zhou, "Rethinking reconstruction autoencoder-based out-of-distribution detection," in *Conference on Computer Vision and Pattern Recognition*, pp. 7379–7387, 2022. 3

[79] N. Durasov, N. Dorndorf, H. Le, and P. Fua, "Zigzag: Universal sampling-free uncertainty estimation through two-step inference," *Transactions on Machine Learning Research*, 2024.

[80] N. Durasov, D. Oner, J. Donier, H. Le, and P. Fua, "Enabling uncertainty estimation in iterative neural networks," in *International Conference on Machine Learning*, 2024. 3

[81] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006. 3

[82] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, and L. Bottou, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion.," *Journal of machine learning research*, vol. 11, no. 12, 2010.

[83] A. Makhzani, J. Shlens, N. Jaitly, and I. Goodfellow, "Adversarial autoencoders," in *International Conference on Learning Representations*, 2016. 3

[84] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. 3

[85] Y. Burda, R. Grosse, and R. Salakhutdinov, "Importance weighted autoencoders," *arXiv preprint arXiv:1509.00519*, 2015.

[86] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework.," *ICLR (Poster)*, vol. 3, 2017. 3

[87] C. J. Maddison, A. Mnih, and Y. W. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," *arXiv preprint arXiv:1611.00712*, 2016. 3, 4

[88] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *International Conference on Learning Representations*, 2017. 4

[89] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, vol. 33, pp. 12449–12460, 2020.

[90] S. Sadhu, D. He, C.-W. Huang, S. H. Mallidi, M. Wu, A. Rastrow, A. Stolcke, J. Droppo, and R. Maas, "Wav2vec-c: A self-supervised model for speech representation learning," *arXiv preprint arXiv:2103.08393*, 2021.

[91] H. Gangloff, M.-T. Pham, L. Courtrai, and S. Lefèvre, "Leveraging vector-quantized variational autoencoder inner metrics for anomaly detection," in *2022 26th International Conference on Pattern Recognition (ICPR)*, pp. 435–441, IEEE, 2022. 3

[92] K. Lis, K. Nakka, M. Salzmann, and P. Fua, "Detecting the Unexpected via Image Resynthesis," in *International Conference on Computer Vision*, 2019. 4

[93] K. Lis, S. Honari, P. Fua, and M. Salzmann, "Detecting Road Obstacles by Erasing Them," in *Transactions on Pattern Analysis and Machine Intelligence*, 2024. 4