Journal of Innovative Image Processing (ISSN: 2582-4252)
www.irojournals.com/iroiip/

Wrapper-Based Adversarial Input Screening for Deep Image Classifiers Using Feature Squeezing and Logit-Space Inconsistency

Alketa Hyso[1], Dezdemona Gjylapi[2]

Department of Computer Science, University "Ismail Qemali", Vlorë, Albania.

E-mail: [1]alketa.hyso@univlora.edu.al, [2]dezdemona.gjylapi@univlora.edu.al

Orcid ID: [1]0009-0000-6256-9831, [2]0009-0005-5186-3636

Abstract

Adversarial perturbations pose a practical integrity risk to vision-based decision systems by causing image classifiers to misclassify inputs after visually subtle changes. This paper evaluates a wrapper-based adversarial input screening approach that compares a classifier's output on an original image with its output after benign feature-squeezing transformations. The evaluated transformations include median filtering, non-local means denoising, and bit-depth reduction. Using a frozen-backbone ResNet50 on CIFAR-10, the detector is assessed under untargeted and targeted Fast Gradient Sign Method attacks, followed by stronger Projected Gradient Descent verification. Detection thresholds are calibrated only on clean data using a fixed 5% false positive rate protocol and are validated on held-out clean samples. The results show that softmax-space $\ell 1$ inconsistency provides moderate, transformation-dependent detection, whereas logit-space $\ell 2$ inconsistency yields a stronger, more stable screening signal. Median filtering with logit-space $\ell 2$ achieves near-complete detection at $\varepsilon = 8/255$ and remains the most reliable configuration across the perturbation sensitivity analysis, while non-local means denoising becomes effective mainly for large perturbations. An additional two-stage gate is evaluated for deployment-oriented alert triage; it does not replace the primary detector or override its screening decision, but ranks flagged inputs by risk severity. Further evaluations show that performance decreases on TinyImageNet and that targeted threshold-aware adaptive optimisation can substantially reduce detection recall. The findings support the use of median filtering with logit-space $\ell 2$ inconsistency as a tool for screening adversarial inputs to image classifiers, but its effectiveness depends on dataset complexity, classifier behaviour, and attack adaptivity.

Keywords: Adversarial Example Detection, Feature Squeezing, Logit-Space Inconsistency, Wrapper-Based Screening, Image Classifier Security.

1. Introduction

Modern information systems increasingly rely on machine learning (ML), particularly deep neural networks, to interpret visual data and drive automated decisions. These systems are widely used in security-oriented applications, including identity verification, access control,

DOI: https://doi.org/10.36548/jiip.2026.3.007
Received: 08.05.2026, received in revised form: 10.06.2026, accepted: 30.06.2026, published: 10.07.2026

perimeter monitoring, anomaly detection, and forensic triage. However, the security assumptions of these pipelines are often violated by adversarial examples: carefully crafted, small-magnitude perturbations to an input image that can cause a trained model to output an incorrect class with high confidence [1]. From an information security perspective, adversarial examples can be viewed as a data integrity attack against the sensing layer [2]. Adversarial perturbation is not like random noise because it is optimised to cross the boundary while remaining visually indistinguishable from the original image. The attacker might exploit this weakness to evade automatic detection, alter labels on surveillance footage, trigger false alarms, or prevent real events from being detected. For example, adversarial accessories can fool face recognition systems [3], and perturbed physical signage can mislead an autonomous vehicle's classifier [4]. This threat is particularly relevant in environments where the model output is used directly for decisions or alerts and where the input channel is exposed to untrusted content.

A direct way to mitigate adversarial risk is to train robust models via adversarial training or to enforce certified robustness, but these approaches can be computationally costly and often reduce standard (clean) accuracy [5]. In fact, robustness may be at odds with accuracy in deep models, as robust training tends to trade off some clean performance for security gains [6]. This trade-off, along with the implementation burden, means that comprehensive retraining or complex verification may not be feasible for legacy systems or rapidly evolving deployments. A number of practical situations can benefit from model-agnostic detector layers that can be applied on top of any pre-trained classifier. Such detectors analyse the model's behaviour when exposed to benign perturbations and measure how inconsistent its output is relative to a calibrated decision threshold. Inputs exceeding this threshold are treated as suspicious and may be filtered, quarantined, or further verified before downstream decisions are made [2].

In this work, adversarial detection is formulated as a practical input screening tool for an image classification pipeline. Before accepting a model's decision as trustworthy, the system performs an inexpensive test to check if the input behaves unusually when benign transformations are applied to it. This formulation of the adversarial detection task makes it a practical pre-screening and warning layer rather than a guarantee of robustness against any adversarial attack. The fundamental idea behind this task is that clean images usually yield consistent model predictions across minor changes introduced by applying a content-preserving transformation, whereas adversarial examples designed to exploit vulnerable decision boundaries become unstable after such transformations [7]. This detection strategy is based on the idea of feature-squeezing defences, where the input is subjected to various transformations, such as spatial smoothing, denoising, or a decrease in colour bit depth, to suppress the effects of adversarial perturbations to detect model prediction instability [7, 8]. According to previous studies, clean inputs usually provide stable predictions under these transformations, but adversarial examples often cause a large output distribution shift [7, 9]. It becomes possible to use the explicit measurement of the discrepancy between the predictions of the original image and its transformed variants to determine whether there is any adversarial manipulation without changing the classifier [5].

Our main contributions are:

Scope-aware Security Framing: We formalise adversarial image perturbations as data-integrity attacks on machine-learning-based inference pipelines and position adversarial detection as a practical pre-decision input-screening mechanism for image

classifiers. The proposed approach is framed as an alerting and triage aid rather than as a general robustness guarantee.

Wrapper-based Detection Method: We implement a lightweight feature-squeezing detector based on median filtering, non-local means denoising, and bit-depth reduction, using logit-space $\ell 2$ as the primary inconsistency score and softmax-$\ell 1$ as a baseline.

Operational Calibration: We introduce a fixed-5%-FPR calibration procedure based only on clean samples via 95th-percentile thresholding, enabling predictable alert rates for both individual and joint detectors.

Empirical Evaluation: We evaluate the screening layer on CIFAR-10 using Fast Gradient Sign Method (FGSM) attacks at $\epsilon = 8/255$, with sensitivity analysis over multiple ϵ values and additional verification under a stronger Projected Gradient Descent attack (PGD-10, 20, and 40) at the same operating point.

Deployment-oriented Outputs: The implementation exports calibrated thresholds, performance summaries (including realised clean flag rates), and diagnostic plots comparing clean versus adversarial score distributions to support reproducibility and integration into monitoring/auditing workflows.

2. Related Work

Adversarial robustness has been widely studied because small, carefully crafted perturbations can cause deep neural networks to misclassify images while remaining visually subtle to human observers [10], [11], [12]. From a security perspective, such perturbations are commonly interpreted as test-time evasion attacks that compromise the integrity of machine-learning-based inference systems [11], [13], [14]. Physical and real-world demonstrations further show the practical relevance of this threat: adversarial accessories can mislead face-recognition systems [3], while perturbed traffic signs can fool visual classifiers in safety-critical settings [4], [15]. These findings have motivated an active line of research on adversarial attacks and defences in deep learning [10].

The existing defences can be classified as robust training, robustness certification, input transformations, and detection techniques. The adversarial training defence enhances robustness by incorporating adversarial examples during training [16]; the min-max formulation proposed by Madry et al. is commonly cited as the benchmark for empirical robustness [17]. Research has also been conducted on regularising models to improve robustness and smoothness under perturbations [18]. Nevertheless, the robustness-driven training process can be quite resource-intensive and negatively affect clean accuracy [19], [6]. Robustness certification guarantees a bound against perturbations; yet, these defences tend to be more complicated to implement within lightweight inference workflows [20].

Input-transformation defences offer a more practical alternative when retraining is costly. Feature squeezing reduces unnecessary input complexity through operations such as colour-depth reduction, spatial smoothing, compression, denoising, and related input perturbation or reconstruction-based transformations [7], [1], [2], [21]. The key intuition is that clean images should preserve relatively stable predictions under benign transformations, whereas adversarial examples often induce larger output changes [7], [8]. This idea also

supports adversarial detection: instead of modifying the classifier, a wrapper-based detector compares the model's response before and after transformation and flags inputs with unusually high inconsistency [7]. Related approaches include sensitivity-inconsistency methods [8], MagNet-style reconstruction and detection mechanisms [22], and Bayesian or uncertainty-based adversarial detection [5].

Despite their practical appeal, transformation-based and inconsistency-based detectors remain sensitive to calibration, transformation choice, dataset complexity, and adaptive attacks. Prior work has shown that maintaining high detection performance while keeping false-positive rates low can be difficult, especially when attackers are aware of the detector [23]. Therefore, a deployment-oriented detector must be evaluated not only by detection accuracy, but also by its behaviour under a fixed operational false-positive-rate constraint. This study follows that perspective by comparing softmax-space and logit-space inconsistency scores under clean-calibrated 5% FPR thresholds, and by testing the resulting screening layer under FGSM, PGD, adaptive, transfer, distortion, and scalability settings.

3. Methodology

3.1 Dataset and Preprocessing

The main experiments were conducted on CIFAR-10, a standard image-classification benchmark containing 60,000 RGB images, 32×32 pixels in size, distributed across 10 classes, with 50,000 training images and 10,000 test images [24]. CIFAR-10 was used as the primary benchmark because it provides a controlled setting for evaluating adversarial perturbations and feature-squeezing-based detection. A separate TinyImageNet experiment was included solely as an additional scalability check on a larger, more visually diverse dataset.

All images were resized to 224×224 pixels to ensure compatibility with ImageNet-pretrained architectures and were normalised using the standard ImageNet statistics:

$$\mu = [0.485, 0.456, 0.406], \quad \sigma = [0.229, 0.224, 0.225] \tag{1}$$

For CIFAR-10 training, we used a lightweight probabilistic augmentation strategy: with a probability of 0.2, each image was processed using one randomly selected spatial transformation from random resized crop, horizontal flip, or vertical flip, followed by colour jittering. Validation and test samples were processed without augmentation, using only resizing and normalisation. Vertical flipping was used only within this low-probability CIFAR-10 augmentation block and was removed from the TinyImageNet experiment because it may be less semantically natural for many object categories.

Experiments were implemented in PyTorch and executed in Google Colab with GPU acceleration using an NVIDIA A100 GPU. CUDA-compatible PyTorch and TorchVision builds were used, and the exact software versions were recorded at runtime for reproducibility.

3.2 Baseline Classifier: Architecture and Training Strategy

The baseline classifier is a ResNet50 model initialised with ImageNet-pretrained weights. Its final fully connected layer is replaced with a 10-class output layer for CIFAR-10. To emulate a lightweight transfer-learning deployment and reduce computational cost, the convolutional backbone is frozen, with only the final classification layer trained.

Training is performed for 10 epochs using the Adam optimiser with a learning rate of 0.009 and weight decay of 1×10−9. The checkpoint with the highest held-out CIFAR-10 evaluation accuracy is selected and kept fixed for all subsequent adversarial attack and detection experiments. In this way, the proposed detector is evaluated as an external screening wrapper around a fixed deployed classifier.

3.3 Threat Model and Attack Generation

We model adversarial examples as test-time data-integrity attacks on the inference pipeline. The adversary is assumed to have white-box access to the classifier parameters and gradients and is constrained by an $\ell\infty$ perturbation budget ϵ

For the untargeted FGSM attack, given an input image x, its true label y, model parameters θ, and loss function $\mathcal{L}$, the untargeted FGSM perturbation is defined as:

$$x_{\text{adv}} = x + \epsilon \cdot \text{sign}\big(\nabla_x L(f_\theta(x), y)\big) \tag{2}$$

The default perturbation budget is $\epsilon = 8/255$ and is performed over $\epsilon \in \{2/255,\ 4/255,\ 8/255,\ 16/255\}$. For targeted FGSM, the target class is selected automatically as the second-most-likely class predicted for the clean input:

$$y_t = arg\ top_2\big(f_\theta(x)\big). \tag{3}$$

The targeted perturbation is then computed to increase the model's confidence in y_t , yielding adversarial inputs that aim for controlled misclassification toward this selected target.

To verify that the results are not limited to single-step attacks, we also evaluate PGD, a stronger multi-step ℓ_∞-bounded attack. Starting from the clean input x_0, PGD iteratively refines an adversarial candidate $x^{(k)}$ by taking gradient-sign steps and projecting the result back into the ℓ_∞ ball of radius ϵ centered at x_0:

$$x^{(k+1)} = \Pi_{||x-x_0||_\infty \le \epsilon}\left(x^{(k)} + \alpha \cdot \text{sign}\left(\nabla_x \mathcal{L}\big(f_\theta\big(x^{(k)}\big), y\big)\right)\right). \tag{4}$$

PGD is evaluated with ϵ = 8/255, step size α = 2/255, random initialisation within the ϵ-ball, and we evaluate k ∈ {10, 20, 40} iterations. For both FGSM and PGD, perturbations are applied consistently in the normalised input space using channel-wise scaling by σ, and adversarial samples are clipped so that the corresponding de-normalised images remain within the valid pixel range [0, 1].

3.4 Feature Squeezing Transformations

Adversarial screening is implemented using feature squeezing, which applies benign input transformations intended to attenuate small, adversarial perturbations while preserving semantic content. Each transformation is applied in pixel space [0, 1], and the resulting image is then renormalised before being passed to the classifier.

Three kinds of operators for image squeezing are studied: median filtering with a radius-1 footprint (approximately 3×3 support), which helps reduce localised high-frequency noise; non-local means (NLMeans) denoising, which is used for structure-preserving filtering based on self-similarity in fast mode; and bit-depth reduction, which quantises pixel intensities to

lower precision. While bit-depth reduction is retained for comparison, the study focuses primarily on median filtering and NLMeans.

3.5 Detection Score: From Softmax-ℓ_1 to Logits-ℓ_2 Distance

We first consider a standard feature squeezing inconsistency score defined as the $\ell 1$ distance between softmax output distributions before and after squeezing:

$$\Delta_{\text{soft}}(x) = ||\text{softmax}\big(f(x)\big) - \text{softmax}\left(f\big(S(x)\big)\right)||_1 \quad (5)$$

Here, x denotes an input image; S(·) is a feature squeezing transformation (e.g., median filtering, NLMeans denoising, or bit-depth reduction); and f(·) denotes the classifier's pre-softmax output vector for a given input. The operator softmax(·) converts this output into a probability distribution over the C classes, and $\| \cdot \|1$ denotes the $\ell 1$ norm over the resulting C-dimensional probability vectors.

However, such a score might suffer from saturation and lack of separability under certain transformations. To improve discriminative power, we adopt an alternative score computed in logit space (pre-softmax), using the $\ell 2$ distance between logits:

$$\Delta_{\text{logit}}(x) = ||\text{z}(x) - \text{z}\big(S(x)\big)||_2 \quad (6)$$

where z(·) denotes the model's logit vector (pre-softmax scores) and $\| \cdot \|2$ denotes the Euclidean norm in R^C. Unless stated otherwise, we use logits-$\ell 2$ as the primary detection score, as it provides substantially stronger separation between clean and adversarial inputs in our experiments.

3.6 Primary Detector Calibration and Optional Alert Triage

Thresholds are calculated using the training samples to maintain a controlled level of false positives within operations. For each detection score $\Delta(\cdot)$, we set the threshold τ to the 95th percentile of the clean-score distribution:

$$\tau = Q_{0.95}(\,\Delta(x) \mid x \sim \text{clean}\,) \quad (7)$$

An input is flagged as suspicious if $\Delta(x) > \tau$, targeting an operational false-positive rate (FPR) of 5% on clean data. To assess whether this operating point generalises beyond the calibration set, we also report the realised flag rate (estimated FPR) on held-out clean validation samples.

When using joint detectors, two scores are combined via a max operator:

$$\Delta_{max}(x) = \max\big(\Delta_a(x), \Delta_b(x)\big) \quad (8)$$

and the corresponding joint threshold is calibrated on clean samples using the same 95th-percentile procedure.

For deployment screening, we use a two-step gate mechanism. The first step involves median squeezing and computing the logits-$\ell 2$ score using the clean, calibrated threshold τ_primary. Inputs greater than τ_primary are flagged as suspicious. The second step is applied to these suspicious inputs and involves a lightweight stability test using benign transformations

such as downsample–upsample resampling, a mild Gaussian blur, and small changes in brightness and contrast. For each flagged input, we compute three instability signals: the label-flip rate, the average confidence drops for the original top-1 label, and the reduction in the top-1 versus top-2 probability margin. The confirmatory score is the maximum of these three quantities.

The confirmatory stage is used only for alert triage, separating flagged inputs into medium- and high-risk alerts; it does not override the primary detection decision. Figure 1 summarises the primary adversarial input screening procedure and the optional second-stage alert-triage process applied only to flagged inputs. The two-stage gate is not used to replace the primary detection rule. Its role is limited to deployment-oriented alert triage after the primary detector has already flagged an input as suspicious.

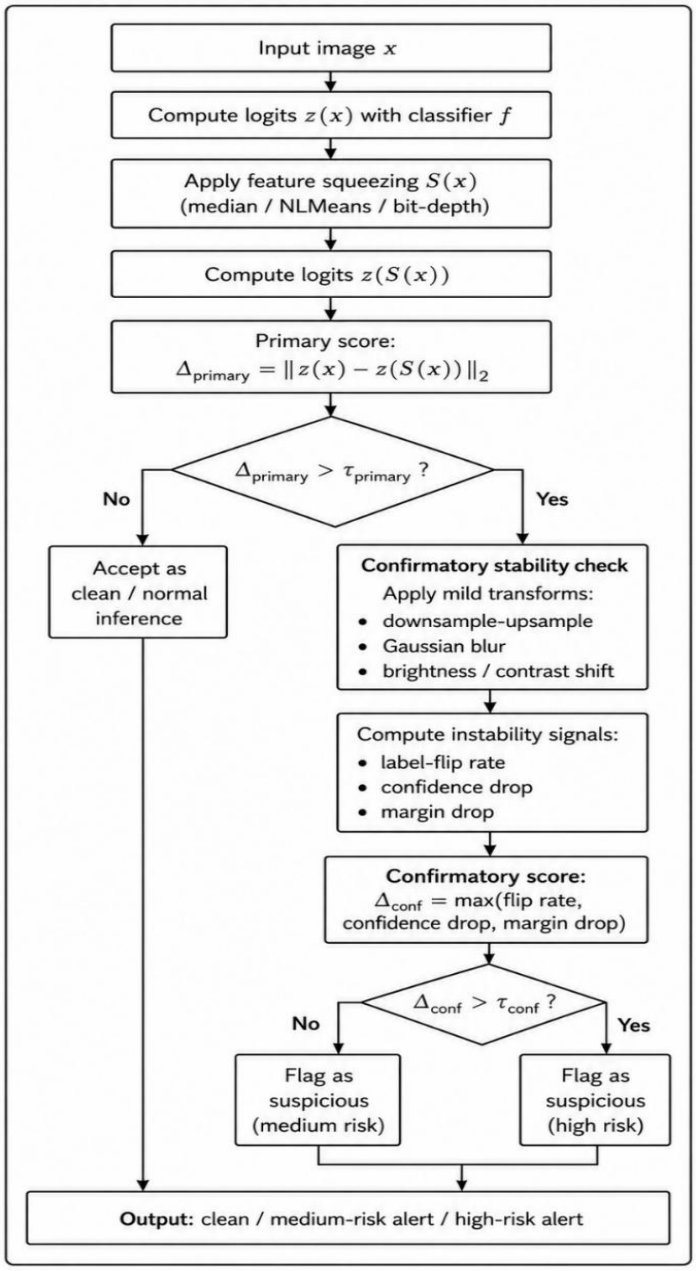

Figure 1. Block diagram of the primary adversarial input detector and optional two-stage alert-triage procedure.

3.7 Reconstruction-Based Comparative Baseline

To provide a reconstruction-based external comparator for the benchmark reported in Section 4.11, we implemented a simplified MagNet-style baseline. In Table 11, this baseline is reported as "MagNet-style (AE recon MSE)". The baseline uses a convolutional autoencoder operating on de-normalised pixel-space images scaled to the range [0, 1]. The encoder contains four strided convolutional layers with channel progression $3 \rightarrow 32 \rightarrow 64 \rightarrow 128 \rightarrow 128$. Each convolution uses a kernel size of 4, a stride of 2, and padding of 1, followed by a ReLU activation. The decoder mirrors this structure using four transposed-convolution layers with channel progression $128 \rightarrow 128 \rightarrow 64 \rightarrow 32 \rightarrow 3$, followed by a final sigmoid output layer to produce a 3-channel RGB reconstruction. In this way, the encoder increases the channel depth from 3 to 128 over four layers and maintains a depth of 128 in the deepest layer, while the decoder maps the latent representation back to image space.

The autoencoder is trained from scratch on a fixed subset of 10,000 clean CIFAR-10 training images resized to 224 × 224 pixels. To maintain consistency with the detector evaluation pipeline, the reconstruction baseline is trained on clean evaluation-format images rather than augmented samples. Training is performed using the Adam optimiser with a learning rate of 1×10^{-3}, weight decay of 1×10^{-6}, a batch size of 32, and 5 training epochs. The objective is to minimise the pixel-wise mean-squared reconstruction error between the original and reconstructed images.

The autoencoder is used only as an external reconstruction-based comparison method and does not modify the protected classifier. For each normalised input, the image is first converted back to pixel space and then reconstructed by the autoencoder. The detection score is defined as the mean squared reconstruction error between the original de-normalised image and its reconstructed output. Higher reconstruction-error values indicate greater deviation from the clean-image reconstruction pattern and are therefore treated as more suspicious.

For fair comparison with the proposed screening method, the reconstruction threshold is calibrated using the same clean calibration protocol as the main detector. Specifically, reconstruction scores are computed on 1,280 clean training samples, and the reconstruction threshold is set to the 95th percentile of the clean-score distribution, targeting a fixed 5% false positive rate. The realised false positive rate (FPR) is then evaluated on held-out clean data.

Adversarial detection is measured on the same untargeted FGSM, targeted FGSM with the per-sample second-most-likely clean class as the target, and untargeted PGD with 20 iterations used in the comparative benchmark. These adversarial examples are generated against the fixed baseline classifier and are then scored by the reconstruction module. We refer to this comparator as MagNet-style because it captures MagNet's reconstruction-based detection component; however, it does not implement a full MagNet reformer-detector ensemble.

3.8 Evaluation Protocol and Metrics

The evaluation is designed based on three main aspects: attack effectiveness, screening effectiveness, and deployment behaviour. The main experiments use FGSM at $\epsilon = 8/255$ with sensitivity analysis over $\epsilon \in \{2/255, 4/255, 8/255, 16/255\}$. To demonstrate that the results are not specific to one-step perturbations, experiments with PGD attacks of 10, 20, and 40 iterations at the same perturbation budget are also performed. Besides the main white-box

experiment, BPDA-lite attacks, black-box transfer attacks, real-world non-adversarial transformations, scalability for TinyImageNet, additional tests on architectures and training methods, and a basic MagNet reconstruction approach are included.

The model's clean accuracy is measured as the baseline classifier's accuracy on clean input data. In the case of untargeted FGSM and PGD attacks, the attack performance is measured using the fooling rate which is the probability of an adversarial example being misclassified, $P(\arg\max f\,(x_{\text{adv}}) \neq y)$. For targeted attacks, the targeted success rate is used to measure performance, which is the probability of an adversarial example being classified as the targeted class, $P(\arg\max f\,(x_{\text{adv}}) = y_t)$, where y_t is the intended target label. The screening performance will be evaluated using the detection rate or true positive rate at the fixed 5% FPR operating point calculated as the fraction of adversarial examples detected by $\Delta(x_{adv}) > \tau$. Where relevant, we also report ROC-AUC, precision, recall, F1-score, bootstrap confidence intervals, missed successful attacks, realised FPR, and runtime overhead relative to the baseline classifier inference.

4. Evaluation and Results

4.1 Baseline Classifier Performance

The ImageNet-pretrained ResNet50 baseline (frozen backbone, only the final classification layer fine-tuned) converges within 10 epochs on clean CIFAR-10. As shown in Figure 2, the held-out evaluation accuracy peaks at 0.7973. We select this best checkpoint and keep it fixed for all subsequent adversarial attack generation and detection experiments, ensuring that later results reflect the screening layer rather than changes in the classifier. This fixed baseline represents the deployed classifier, while the detector is evaluated as an external screening wrapper.

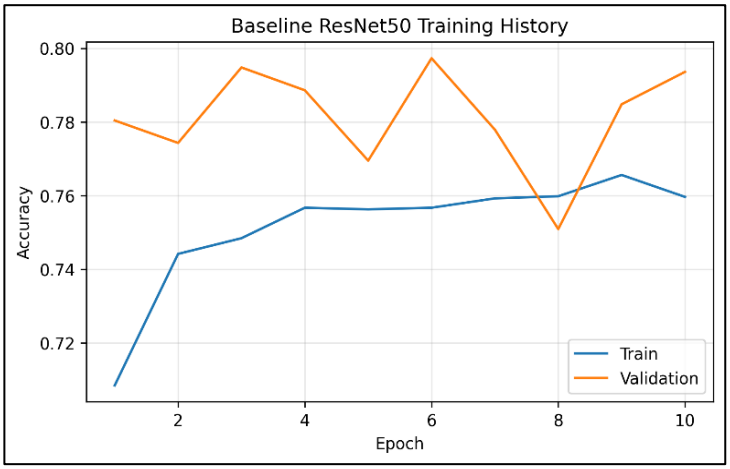

Figure 2. Training and evaluation accuracy of the ImageNet-pretrained ResNet50 baseline on CIFAR-10 over 10 epochs (frozen backbone, final layer fine-tuned).

4.2 FGSM Attack Effectiveness and Qualitative Examples

The baseline model's vulnerability at our chosen perturbation budget is evaluated by generating adversarial examples via FGSM with $\epsilon = 8/255$ (i.e., $\epsilon = 0.03137$ in pixel space). This is done for both untargeted FGSM, which aims to cause any misclassification, and targeted

FGSM, where the target is set to the second-most-probable class according to the original prediction.

On a representative validation batch, untargeted FGSM achieves a fooling rate of 0.9313, measured as the fraction of samples whose adversarial predictions differ from their true labels. In contrast, targeted FGSM toward the per-sample second-most-likely label attains a success rate of 0.0646, where success is defined strictly as the adversarial prediction matching the selected target. Intuitively, it means that at this operating point, one-step attacks are usually enough to disrupt the classifier (fooling it into any misclassification), but it is much more difficult to force a particular class (i.e., succeed in the chosen target). Representative examples are shown in Figures 3 and 4 as Original | Fooled | Difference, while the reported fooling and success rates are computed directly from the validation batch.

Reproducibility details: Attacks are generated in the model's normalised input space: the $\ell\infty$ budget ϵ is converted channel-wise via division by the ImageNet standard deviation (ϵnorm = ϵ/σ), the FGSM update is applied using the sign of the input gradient, and outputs are clamped so that the corresponding de-normalised images remain within valid pixel bounds [0,1]. Reported fooling/success rates are computed directly from the model's top-1 predictions on the clean batch versus the adversarial batch, using the definitions above.

The "Fooled" images are adversarially perturbed inputs that look nearly identical to the originals at $\epsilon = 8/255$, yet can still induce a different model prediction; the "Difference" panel visualises these otherwise imperceptible pixel-level changes (contrast-enhanced). Although the perturbation is visually subtle at $\epsilon = 8/255$, it can still produce a decisive change in the model's predicted label, illustrating the practical risk of small, input-level integrity attacks in a deployed inference pipeline.

Figure 3. Untargeted FGSM examples at $\epsilon = 8/255$: Original image, adversarial image, and contrast-enhanced perturbation map.

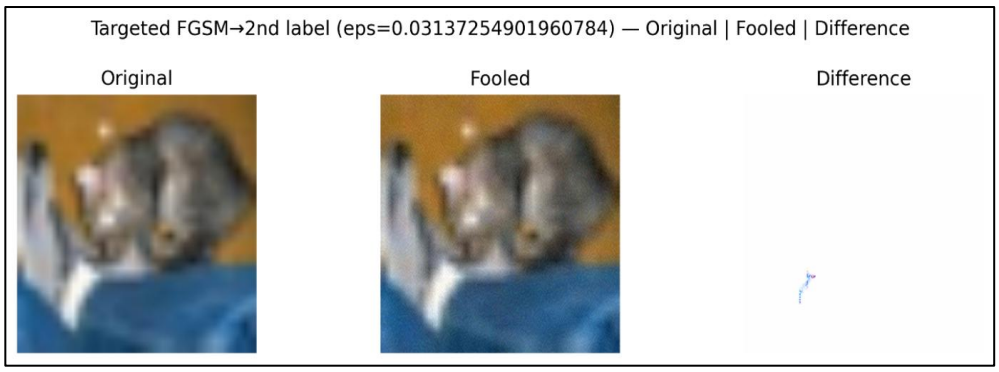

Figure 4. Targeted FGSM (target = second-most-likely clean label) at $\epsilon = 8/255$: Original image, adversarial image, and contrast-enhanced perturbation map.

The relatively low targeted FGSM success rate is expected in this setting because the attack is limited to a single gradient step at $\epsilon = 8/255$ and must force the prediction toward a specific per-sample target class. This is substantially more difficult than causing any misclassification. The targeted FGSM results should therefore be interpreted as reflecting detector behaviour under a relatively weak targeted single-step attack, rather than as a general statement about robustness to targeted attacks.

4.3 From Softmax-ℓ1 to Logits-ℓ2 at Fixed 5% FPR

At the same fixed operating point (5% FPR), we compare two feature-squeezing inconsistency scores: the standard softmax-ℓ1 score and the alternative logits-ℓ2 score. This comparison isolates the effect of the scoring function while keeping the calibration protocol, evaluation subset, and squeezing transformations unchanged.

As shown in Table 1, the standard softmax-ℓ1 score yields only moderate, transformation-dependent detection. Median filtering performs best among the individual detectors, followed by NLMeans, whereas bit-depth reduction performs poorly. Joint max-fusion does not improve performance over median filtering alone, suggesting that the softmax-ℓ1 score is not a sufficiently stable screening signal at this low-FPR operating point.

Table 1. Detection at 5% FPR (Softmax–ℓ1, FGSM, $\epsilon = 8/255$; Untargeted vs. Targeted)

Detector	Type	Threshold τ (5% FPR)	Detection rate (untargeted)	Detection rate (targeted)
median	individual	1.4750	0.8750	0.8708
nlmeans	individual	1.9931	0.5688	0.5479
bitdepth	individual	1.9958	0.0000	0.0000
max(median,nlmeans)	joint	1.9931	0.6604	0.6375
max(median,bitdepth)	joint	1.9958	0.3104	0.2875
max(nlmeans,bitdepth)	joint	1.9982	0.4063	0.4042

Table 2 shows that replacing softmax-ℓ1 with logits-ℓ2 substantially improves detection under the same calibration protocol. Median filtering becomes the most reliable primary detector, achieving near-complete detection for both untargeted and targeted FGSM. NLMeans also improves considerably, whereas bit-depth reduction remains ineffective. Since joint max-fusion does not outperform the median detector, the improvement is mainly due to the logit-space discrepancy rather than detector combination. Overall, median filtering with logits-ℓ2 provides the strongest and most consistent screening configuration for the subsequent experiments.

Table 2. Detection at 5% FPR (logits-ℓ2, $\epsilon = 8/255$, N = 480)

Detector	Type	Threshold τ (5% FPR)	Detection rate (untargeted)	Detection rate (targeted)
median	individual	11.9803	0.9958	0.9958
nlmeans	individual	27.1790	0.9563	0.9521
bitdepth	individual	31.7187	0.0000	0.0000
max(median,nlmeans)	joint	27.1790	0.9813	0.9792
max(median,bitdepth)	joint	31.7187	0.9500	0.9438
max(nlmeans,bitdepth)	joint	32.2203	0.8750	0.8688

4.4 Operating-Point Validation for the Logits-ℓ2 Detector

To validate the operating point used in the logits-ℓ2 detector comparison, we examined threshold stability, held-out clean false-positive behaviour, ROC-based separability, and

bootstrap confidence intervals. The analysis focuses on the strongest individual detector, median + logits-ℓ2, and on the individual logits-ℓ2 feature-squeezing detectors evaluated at $\epsilon = 8/255$. First, we assessed the sensitivity of threshold calibration to the number of clean calibration samples. As shown in Table 3, the threshold remains stable across Nclean = 320, 640, 1280, 2560, and 5000. The realised FPR on held-out clean validation data remains close to the nominal 5% target, while detection remains unchanged at 0.9958 for both untargeted and targeted FGSM. These results indicate that Nclean = 1280 provides a stable and practical calibration setting for the main CIFAR-10 experiments.

To complement the detector-comparison results reported in Table 2, we additionally report ROC-AUC, precision, recall, F1-score, and realised FPR for the individual logits-ℓ2 detectors under the same fixed-FPR evaluation setting.

Table 3. Threshold stability analysis for the median + logits-ℓ2 detector under FGSM at $\epsilon = 8/255$.

Detector	Nclean	Threshold τ (95th percentile)	Realised FPR on held-out clean validation	Detection rate (untargeted)	Detection rate (targeted)
median + logits-ℓ2	320	11.9946	0.0480	0.9958	0.9958
median + logits-ℓ2	640	11.9581	0.0486	0.9958	0.9958
median + logits-ℓ2	1280	11.9803	0.0482	0.9958	0.9958
median + logits-ℓ2	2560	11.8990	0.0502	0.9958	0.9958
median + logits-ℓ2	5000	11.8077	0.0524	0.9958	0.9958

Because these metrics are evaluated against held-out clean validation samples, small numerical differences relative to Table 2 are expected, although the qualitative detector ranking remains unchanged.

Table 4. Additional evaluation metrics for logits-ℓ2 feature-squeezing detectors at $\epsilon = 8/255$.

Detector	Attack	ROC-AUC	Precision	Recall	F1-score	FPR
median-ℓ2	untargeted FGSM	0.9995	0.6648	0.9958	0.7973	0.0482
median-ℓ2	targeted FGSM	0.9995	0.6648	0.9958	0.7973	0.0482
NLMeans	untargeted FGSM	0.9889	0.6245	0.9563	0.7556	0.0552
NLMeans	targeted FGSM	0.9877	0.6235	0.9521	0.7535	0.0552
bit-depth	untargeted FGSM	0.0000	0.0000	0.0000	0.0000	0.0590
bit-depth	targeted FGSM	0.0000	0.0000	0.0000	0.0000	0.0590

As shown in Table 4, median + logits-ℓ2 remains the strongest detector, with near-perfect separability and a recall of 0.9958 for both untargeted and targeted FGSM. NLMeans is weaker but still effective, while bit-depth reduction shows no useful separation between clean and adversarial examples. The identical median-detector metrics across the two FGSM settings reflect that almost all adversarial samples exceed the same clean-calibrated threshold.

This ranking is further illustrated by the ROC curves in Figure 5 for the logits-ℓ2 detectors under untargeted and targeted FGSM at $\epsilon = 8/255$.

Finally, we estimated 95% bootstrap confidence intervals using 1,000 resamples. For median + logits-ℓ2, ROC-AUC remained extremely high in both settings: 0.9995 for untargeted FGSM, with 95% CI [0.9986, 1.0000], and 0.9995 for targeted FGSM, with 95% CI [0.9986, 1.0000]. Recall also remained stable at 0.9958, with 95% CI [0.9875, 1.0000] for untargeted FGSM and [0.9896, 1.0000] for targeted FGSM. NLMeans showed slightly lower ROC-AUC and recall, while bit-depth reduction remained ineffective. Overall, the operating-point validation confirms that median + logits-ℓ2 provides the most stable and reliable detector configuration under the fixed 5% FPR protocol.

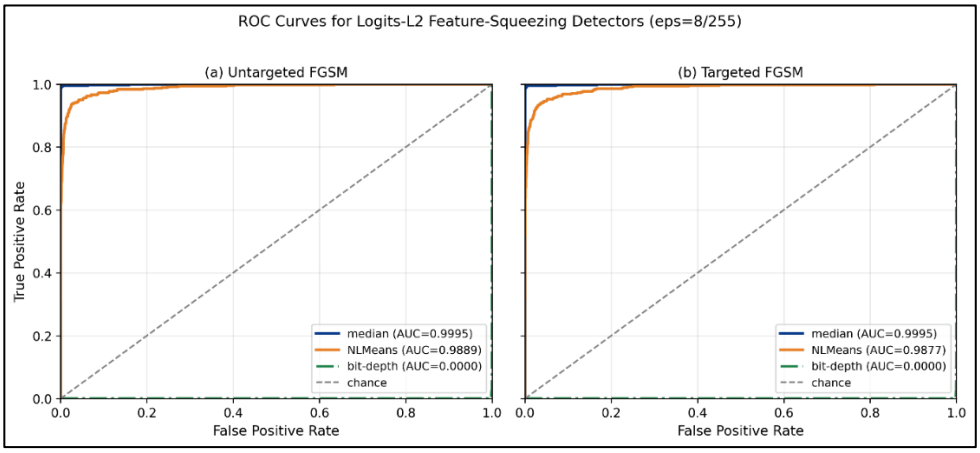

Figure 5. ROC curves for the logits-ℓ2 feature-squeezing detectors under (a) untargeted FGSM and (b) targeted FGSM at $\epsilon = 8/255$.

4.5 Two-Stage Screening Gate for Deployment

To examine whether the strongest detector can be deployed as a practical pre-decision screening layer, we evaluated a two-stage gate built on median + logits-ℓ2. The primary stage uses the same clean-calibrated detector as in the main CIFAR-10 experiments, while the second stage performs a lightweight confirmatory stability check for alert triage only. The final screening decision remains determined by the primary detector, and the confirmatory stage is used only to separate flagged inputs into medium-risk and high-risk alerts.

As shown in Table 5, the two-stage gate preserves the strong detection performance of the primary median + logits-ℓ2 detector while maintaining a held-out clean FPR close to the intended 5% operating point. In both untargeted and targeted FGSM settings, the detection rate remains 0.9958 because the confirmatory stage does not override the primary decision. Instead, it adds operational value by ranking flagged inputs according to alert severity.

Table 5. Two-stage screening gate results for the median + logits-ℓ2 detector at $\epsilon = 8/255$.

Attack	τprimary	τconf	Held-out clean FPR	Detection rate	High-risk among flagged
FGSM untargeted	11.9803	0.5102	0.0469	0.9958	0.1925
FGSM targeted	11.9803	0.5102	0.0469	0.9958	0.1987

In this setting, 19.25% of flagged untargeted adversarial inputs and 19.87% of flagged targeted adversarial inputs are assigned to the high-risk tier, while the remaining flagged cases are treated as medium-risk alerts. This supports the use of the two-stage design as a practical screening wrapper that preserves detection strength while enabling more actionable alert prioritisation.

4.6 Sensitivity to ϵ (FGSM, logits-ℓ2; fixed 5% FPR)

We study how the perturbation budget ϵ affects attack effectiveness and detection performance under a fixed operational false-alarm budget. For each detector, the decision threshold is calibrated once on clean training data to enforce a 5% FPR (the 95th percentile of clean scores). We then evaluate FGSM on validation data over 15 batches (batch size = 32, N = 480 samples) for each $\epsilon \in \{2/255, 4/255, 8/255, 16/255\}$. Targeted FGSM uses the per-sample second-most-likely clean class as the target. Table 6 reports the untargeted fooling rate and targeted success rate across perturbation budgets.

Median filtering with logits-ℓ2 provides robust coverage across perturbation budgets: detection increases from 0.7313 at 2/255 to 0.8271 at 4/255, reaches 0.9958 at 8/255, and becomes perfect at 16/255 for both untargeted and targeted FGSM.

Table 6. FGSM attack effectiveness across perturbation budgets ϵ: untargeted fooling rate and targeted success rate (target = second-most-likely clean class).

ϵ	FGSM fooling rate (Untargeted)	FGSM targeted success rate
2/255	0.8438	0.1417
4/255	0.8875	0.0458
8/255	0.9313	0.0646
16/255	0.9083	0.1333

In contrast, NLMeans exhibits sharp "turn-on" behaviour: it remains relatively weak at smaller budgets (0.2542/0.2729 at 2/255, 0.2958/0.3042 at 4/255 for untargeted/targeted FGSM), but improves substantially at larger perturbation magnitudes (0.9563/0.9521 at 8/255 and 1.0/1.0 at 16/255). Bit-depth reduction remains ineffective throughout. Overall, median filtering is the most reliable primary gate when the goal is consistent screening under a fixed false-alarm budget. If a deployment requires a single stable screening rule at 5% FPR, median + logits-ℓ2 is the most dependable choice across ϵ, whereas NLMeans becomes attractive only at larger perturbation magnitudes.

4.7 PGD-10, PGD-20 and PGD-40 Verification

To evaluate the detector under stronger multi-step adversarial attacks, we tested PGD with 10, 20 and 40 iterations. All PGD attacks were generated with $\epsilon = 8/255$, step size $\alpha = 2/255$, and random initialisation. The evaluation was performed on 15 validation batches (N = 480 samples) using the same fixed 5% FPR operating point calibrated on clean data.

As shown in Table 7, all PGD configurations achieved a fooling rate of 1.0000, confirming that the baseline classifier was fully vulnerable to these iterative perturbations. In contrast, median + logits-ℓ2 and the two-stage gate maintained perfect detection across PGD-10, PGD-20 and PGD-40.

Table 7. Detection under stronger PGD attacks at $\epsilon = 8/255$

Attack	Steps	Fooling rate	Median + logits-ℓ2	NLMeans + logits-ℓ2	Bit-depth + logits-ℓ2	Two-stage gate
PGD	10	1.0000	1.0000	0.9000	0.0000	1.0000
PGD	20	1.0000	1.0000	0.9958	0.0000	1.0000
PGD	40	1.0000	1.0000	1.0000	0.0000	1.0000

NLMeans also performed strongly, improving from 0.9000 under PGD-10 to 1.0000 under PGD-40. Bit-depth reduction remained ineffective, with a detection rate of 0.0000 in all settings. This is likely because quantisation affects both clean and adversarial images similarly and does not significantly alter the model's logit responses. Unlike median filtering and NLMeans, it does not explicitly smooth local spatial artifacts or suppress high-frequency perturbation patterns. Therefore, bit-depth reduction is retained only as a negative comparison baseline, while median + logits-ℓ2 is used as the primary screening configuration.

4.8 Strengthened Adaptive Attack Evaluation

To strengthen the adaptive evaluation, we extended the original detector-aware test with two stronger BPDA-lite settings: an untargeted PGD-50 attack and a targeted PGD-50 attack targeting the per-sample second-most-likely clean class. In both adaptive settings, the objective

jointly maximises attack success and minimises the median + logits-ℓ2 detector score at the fixed 5% FPR operating point.

Table 8. Strengthened adaptive evaluation of the median + logits-ℓ2 detector on CIFAR-10 under the fixed-5%-false-positive-rate (FPR) operating point.

Attack	Attack success rate	ROC-AUC	TPR	F1-score	FPR	Threshold @5% FPR	Mean Detector Score
PGD-20 untargeted (detector-unaware)	1.0000	1.0000	1.0000	0.7993	0.0482	11.9803	50.7713
PGD-50 untargeted (detector-unaware)	1.0000	1.0000	1.0000	0.7993	0.0482	11.9803	77.1789
Adaptive PGD-50 untargeted (BPDA-threshold-aware)	1.0000	0.9998	0.9979	0.7983	0.0482	11.9803	39.9838
Targeted PGD-20 (detector-unaware, top-2 target)	0.9833	0.9984	0.9958	0.7973	0.0482	11.9803	24.6113
Adaptive targeted PGD-50 (BPDA-threshold-aware, top-2 target)	0.9792	0.5159	0.0604	0.0773	0.0482	11.9803	7.3255

As shown in Table 8, the detector-unaware untargeted PGD-20 and PGD-50 attacks remain fully detectable, both achieving ROC-AUC = 1.0000 and TPR = 1.0000. The adaptive untargeted PGD-50 attack also remains highly detectable, with ROC-AUC = 0.9998 and TPR = 0.9979. Importantly, this should not be interpreted as evidence that the adaptive objective failed to reduce the detector score. Relative to detector-unaware PGD-50, the adaptive untargeted attack lowers the mean median + logits-ℓ2 detector score from 77.18 to 39.98, while the clean-calibrated decision threshold remains 11.98. Thus, the adaptive objective weakens the score margin and slightly reduces separability, but most successful adversarial samples still remain well above the threshold, so thresholded detection changes only marginally from 1.0 to 0.9979.

Figure 6 illustrates this behaviour at the score-distribution level: the adaptive untargeted distribution shifts leftward relative to detector-unaware PGD-50, yet remains largely to the right of the clean-calibrated threshold. Under the present untargeted objective and perturbation budget, BPDA-lite therefore reduces score magnitude, but not enough to produce widespread threshold evasion.

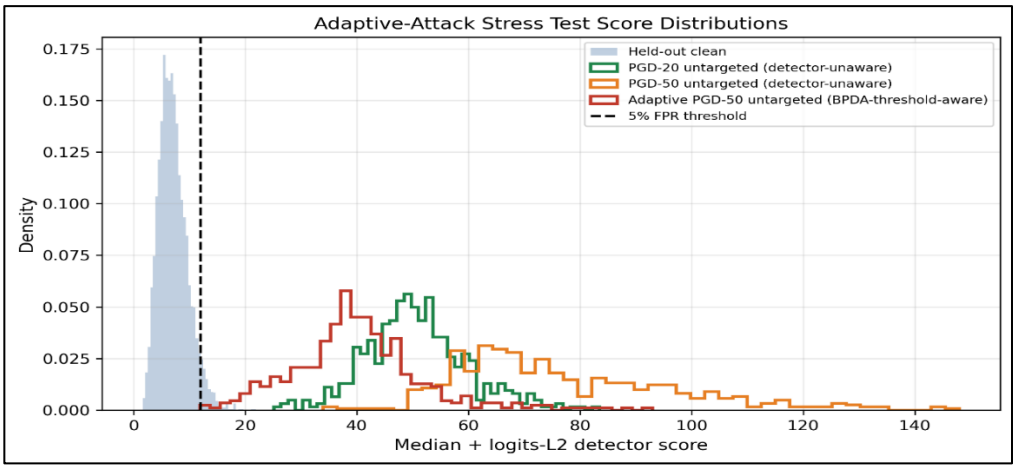

Figure 6. Untargeted adaptive-attack score distributions for the median + logit-space ℓ2 detector at the fixed 5% FPR operating point. Held-out clean, detector-unaware PGD-20, detector-unaware PGD-50, and adaptive untargeted PGD-50 (BPDA-threshold-aware) are shown. The dashed vertical line indicates the clean-calibrated decision threshold.

The targeted setting yields a sharply different result. Although detector-unaware targeted PGD-20 achieves a high attack success rate of 0.9833, it remains well separated from clean data, with ROC-AUC = 0.9984 and TPR = 0.9958. By contrast, the adaptive targeted PGD-50 attack preserves a similarly high attack success rate of 0.9792 while sharply reducing detector separability, with ROC-AUC dropping to 0.5159 and TPR to 0.0604. In this case, the mean detector score falls to 7.33, below the same decision threshold of 11.98, which explains the strong increase in threshold evasion.

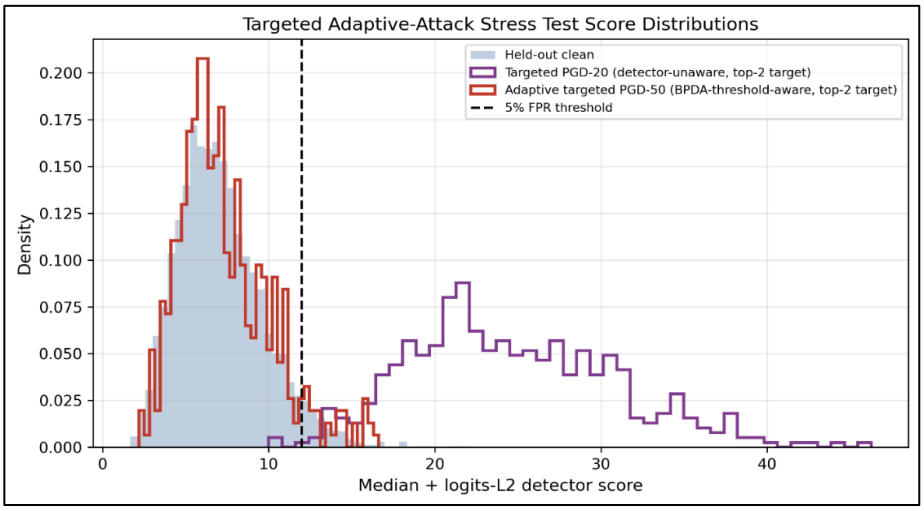

Figure 7. Targeted adaptive-attack score distributions for the median + logit-space $\ell 2$ detector at the fixed 5% FPR operating point. Held-out clean, targeted detector-unaware PGD-20, and adaptive targeted PGD-50 (BPDA-threshold-aware, top-2 target) are shown. The dashed vertical line indicates the clean-calibrated decision threshold.

Figure 7 shows that many successful targeted adaptive adversarial samples shift toward the clean distribution and below the operational decision boundary.

Together, Figures 6 and 7 show that adaptive behaviour is strongly attack-dependent: in the tested untargeted setting, score suppression is insufficient for broad threshold evasion, whereas targeted threshold-aware optimisation can drive many successful attacks below the clean-calibrated threshold. These results highlight an important limitation of the proposed screening layer and motivate further robustness evaluation under stronger detector-aware targeted optimisation.

4.9 TinyImageNet Scalability and Generalisation Check

To assess scalability beyond CIFAR-10, we conducted an additional experiment on TinyImageNet, a more challenging dataset with a larger number of classes and greater visual variability. The same ResNet50-based screening framework was retained, and the evaluation focused on the strongest detector identified in the CIFAR-10 experiments, median + logits-$\ell 2$. The detection threshold was calibrated on 5,000 clean training samples using the same fixed 5% FPR protocol. Vertical flipping was removed from the TinyImageNet augmentation pipeline because it may produce unrealistic transformations for many natural object categories.

Table 9. TinyImageNet scalability and generalisation results for the median + logits-ℓ2 detector at $\epsilon = 8/255$, evaluated under untargeted FGSM and PGD-20.

Attack	Threshold τ	Clean Val. Acc.	ROC-AUC	Precision	TPR	F1-score	FPR
FGSM untargeted	340.1365	0.5651	0.9407	0.9395	0.6146	0.7431	0.0396
PGD-20 untargeted	340.1365	0.5651	0.8879	0.9212	0.4625	0.6158	0.0396

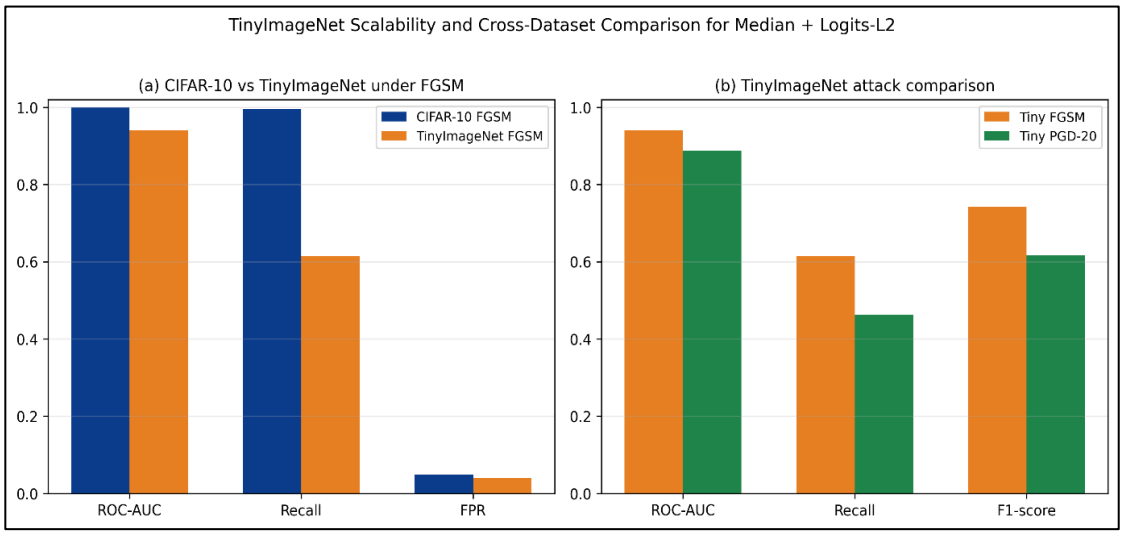

Figure 8. TinyImageNet comparison for the median + logits-ℓ2 detector: (a) CIFAR-10 versus TinyImageNet under untargeted FGSM, shown through ROC-AUC, Recall, and FPR; (b) TinyImageNet performance under untargeted FGSM versus untargeted PGD-20, shown through ROC-AUC, Recall, and F1-score.

Here, TPR denotes the detection rate on adversarial samples, while FPR denotes the realised FPR on held-out clean validation samples. As shown in Table 9, the detector remains informative on TinyImageNet, with ROC-AUC values of 0.9407 for FGSM and 0.8879 for PGD-20. However, the detection rate decreases compared with the main CIFAR-10 experiments, especially under PGD-20, where TPR falls to 0.4625. The degradation is mainly explained by reduced clean/adversarial separability relative to the dataset-specific threshold. On TinyImageNet, the clean score distribution yields a substantially higher calibrated threshold, while many successful adversarial examples do not increase the logits-ℓ2 discrepancy enough to exceed it. As a result, some adversarial inputs remain below the decision boundary even though the classifier is fooled. Figure 8 further illustrates this behaviour by comparing CIFAR-10 and TinyImageNet under untargeted FGSM and by contrasting FGSM with PGD-20 within TinyImageNet.

4.10 Alternative Architecture Check and Training-Strategy Check

To examine whether the reported screening behaviour depends on a single classifier architecture or training regime, we conducted two additional CIFAR-10 checks using the same median and logits-ℓ2 detector at the fixed 5%-FPR operating point. We evaluated DenseNet121 under the same frozen-backbone transfer-learning setting as the baseline ResNet50 and a fully fine-tuned EfficientNet-B0 as an alternative, stronger classifier. The comparative results are reported in Table 10. The additional checks show that the screening signal is not restricted to the original frozen-backbone ResNet50, but its strength depends on the classifier. DenseNet121 achieves slightly higher clean validation accuracy than ResNet50 (0.8062 vs 0.7973), yet detection is clearly weaker. Fully fine-tuned EfficientNet-B0 improves clean validation accuracy to 0.9626, but it does not improve detector

performance, with ROC-AUC remaining around 0.95–0.96 and detection rates around 0.77–0.79. Overall, the framework transfers across architectures, but ResNet50 provides the strongest detector behaviour in this study.

Table 10. Alternative architecture and training-strategy results for the median + logits-ℓ2 detector on CIFAR-10 at $\epsilon = 8/255$.

Model	Backbone frozen	Clean Val. Acc.	Attack	Attack success rate	ROC-AUC	Detection rate	F1-score	FPR
ResNet50	Yes	0.7973	FGSM untargeted	0.9313	0.9995	0.9958	0.7973	0.0482
ResNet50	Yes	0.7973	FGSM targeted	0.0646	0.9995	0.9958	0.7973	0.0482
DenseNet121	Yes	0.8062	FGSM untargeted	0.8063	0.9603	0.7604	0.6919	0.042
DenseNet121	Yes	0.8062	FGSM targeted	0.1604	0.96	0.7833	0.7054	0.042
EfficientNet-B0	No	0.9626	FGSM untargeted	0.8938	0.9595	0.7854	0.7034	0.043
EfficientNet-B0	No	0.9626	FGSM targeted	0.0938	0.9539	0.7688	0.6936	0.043

Figure 9 summarises these architecture-level differences visually, showing that higher clean accuracy does not necessarily translate into stronger detection performance: ResNet50 achieves the strongest untargeted and targeted screening behaviour, while DenseNet121 and EfficientNet-B0 retain useful but lower detection rates.

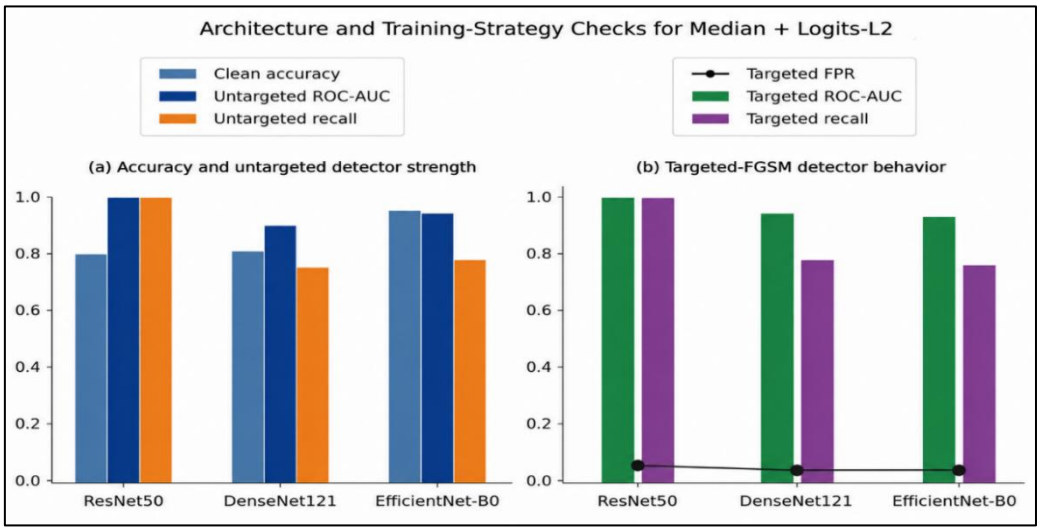

Figure 9. Architecture and training-strategy comparison for the median + logits-ℓ2 detector on CIFAR-10: (a) clean accuracy and untargeted-FGSM screening behaviour across ResNet50, DenseNet121, and EfficientNet-B0; (b) targeted-FGSM screening behaviour across the same models.

4.11 Comparative Benchmark Against Existing Detection Frameworks

To place the proposed screening architecture in context with state-of-the-art adversarial attack detection models, we tested it in the identical 5% FPR setting against two baseline approaches: the classic feature squeezing approach using a median filter with softmax-ℓ1 scoring and the MagNet-inspired reconstruction detector.

Table 11. Comparative benchmark against existing adversarial detection frameworks on CIFAR-10 under the fixed-5%-false-positive-rate (FPR) protocol.

Detector	Attack	Attack success rate	ROC-AUC	TPR	F1-score	FPR
Feature squeezing	FGSM untargeted	0.9313	0.9787	0.8750	0.7317	0.0496
MagNet-style (AE recon MSE)	FGSM untargeted	0.9313	0.9903	0.9979	0.8266	0.0400
Median + logits-ℓ2	FGSM untargeted	0.9313	0.9995	0.9958	0.7973	0.0482
Two-stage gate	FGSM untargeted	0.9313	N/A	0.9958	0.7973	0.0482
Feature squeezing	FGSM targeted	0.0646	0.9780	0.8708	0.7295	0.0496
MagNet-style (AE recon MSE)	FGSM targeted	0.0646	0.9903	0.9979	0.8266	0.0400
Median + logits-ℓ2	FGSM targeted	0.0646	0.9995	0.9958	0.7973	0.0482
Two-stage gate	FGSM targeted	0.0646	N/A	0.9958	0.7973	0.0482
Feature squeezing	PGD-20 untargeted	1.0000	0.8809	0.4979	0.4943	0.0496
MagNet-style (AE recon MSE)	PGD-20 untargeted	1.0000	0.9263	0.2958	0.3455	0.0400
Median + logits-ℓ2	PGD-20 untargeted	1.0000	1.0000	1.0000	0.7993	0.0482
Two-stage gate	PGD-20 untargeted	1.0000	N/A	1.0000	0.7993	0.0482

Table 11 summarises the results. Under untargeted and targeted FGSM at $\epsilon = 8/255$, the proposed median + logits-ℓ2 detector and the two-stage gate both achieve TPR = 0.9958, substantially above the classical softmax-ℓ1 baseline (0.8750 and 0.8708, respectively), while remaining close to the MagNet-style baseline (0.9979 in both settings). The more effective PGD-20 attack results in the following performance difference: the detector based on the median + logits-ℓ2 approach and the two-step gate retain TPR=1.0000, but the traditional softmax-ℓ1 feature squeezing baseline attains 0.4979, while the MagNet-inspired baseline attains 0.2958. A comparable situation occurs with the ROC-AUC metric: while the detector retains a value of 1.0 for the PGD-20 attack, the baselines drop to 0.8809 and 0.9263, respectively. These results show that replacing the classical softmax-space inconsistency score with logits-ℓ2 substantially strengthens the feature-squeezing framework, especially under the stronger iterative PGD-20 attack. The two-stage gate does not improve binary detection beyond the primary detector, but it preserves the primary detector's strong screening performance while adding deployment-oriented alert prioritisation.

4.12 Practical Robustness and Deployment Checks

4.12.1 Black-Box Transfer Evaluation

To evaluate robustness outside the white-box scenario, we conducted an additional experiment transferring black-box attacks from ResNet50 to DenseNet121 on the CIFAR-10 dataset. In the process, adversarial images were created using a source model and tested against a target model. The results are listed in Table 12 below.

Table 12. Black-box transfer evaluation on CIFAR-10.

Source model	Target model	Attack	Source success	Transfer success	Median TPR	NLMeans TPR	Bit-depth TPR	Two-stage TPR
ResNet50	DenseNet121	Untargeted FGSM	0.9313	0.7625	0.9771	0.0958	0.0000	0.9771
ResNet50	DenseNet121	Targeted FGSM	0.0646	0.2125	0.9750	0.0917	0.0000	0.9750
DenseNet121	ResNet50	Untargeted FGSM	0.7979	0.8854	0.4000	0.0500	0.0000	0.4000
DenseNet121	ResNet50	Targeted FGSM	0.1625	0.1104	0.4042	0.0458	0.0000	0.4042

As shown in Table 12, transfer-based detection is strongly asymmetric. Perturbations transferred from ResNet50 to DenseNet121 remain highly detectable by the median detector and the two-stage gate, whereas transfer in the opposite direction is much harder to detect. Successfully transferred attacks from ResNet50 to DenseNet121 still produce detector scores well above the DenseNet121 threshold, while successfully transferred attacks from DenseNet121 to ResNet50 tend to lie much closer to the ResNet50 threshold.

Thus, the asymmetry is not explained solely by transfer success itself, but by how strongly the transferred perturbations activate the target model's squeeze-based detection score relative to its clean-calibrated operating point. NLMeans remains weak in all transfer settings, and bit-depth reduction remains ineffective.

4.12.2 Real-World Distortion Robustness

To examine behaviour under deployment-relevant non-adversarial perturbations, we evaluated the screening layer on several real-world distortions, including resizing, blur, brightness changes, contrast changes, additive Gaussian noise, and JPEG compression. The corresponding results are summarised in Table 13.

As shown in Table 13, the detector remains broadly stable under resizing, blur, moderate brightness shifts, contrast changes, and JPEG compression, with generally low flag rates.

Table 13. Real-world distortion robustness on CIFAR-10.

Distortion	Accuracy	Median flag rate	NLMeans flag rate	Bit-depth flag rate	Two-stage flag rate
clean	0.8094	0.0469	0.0500	0.0547	0.0469
resize down-up	0.7898	0.0055	0.0063	0.0117	0.0055
gaussian blur	0.8141	0.0117	0.018	0.025	0.0117
brightness up	0.8055	0.0516	0.0516	0.0656	0.0516
contrast up	0.7953	0.0422	0.0484	0.0531	0.0422
gaussian noise ($\sigma = 0.01$)	0.3773	0.8656	0.0508	0.0000	0.8656
JPEG quality 75	0.7461	0.0750	0.0578	0.0266	0.0750

Additive Gaussian noise is the main exception: it causes a sharp increase in the flag rate for the median detector and the two-stage gate, together with a substantial drop in classification accuracy. This suggests that high-frequency noise can resemble the instability pattern that the median-based detector associates with adversarial manipulation. In contrast, NLMeans remains comparatively insensitive to this corruption, and bit-depth reduction remains largely unresponsive overall.

4.12.3 Runtime and Computational Overhead

Because the proposed screening layer is intended for deployment, we measured its runtime cost relative to baseline classifier inference on held-out validation batches. The values reported in Table 14 refer to inference-time screening costs, not the full experimental pipeline. The results show a clear trade-off between computational cost and detection value. Median + logits-ℓ2 introduces additional overhead compared with baseline inference, but remains practical as a screening layer. NLMeans provides useful detection performance but is substantially more expensive, making it less suitable for real-time deployment. Bit-depth reduction is computationally cheap, but its weak detection performance limits its practical

value. Overall, median + logits-$\ell 2$ offers the best balance between screening effectiveness and deployment feasibility among the evaluated options.

Table 14. Runtime and computational overhead relative to baseline inference.

Method	Total time (s)	Time per image (s)	Relative overhead
Baseline classifier inference only	0.1494	0.000467	1.00×
Median squeezing + logits-$\ell 2$	2.5558	0.007987	17.11×
NLMeans squeezing + logits-$\ell 2$	23.3465	0.072958	156.31×
Bit-depth squeezing + logits-$\ell 2$	0.4807	0.001502	3.22×

5. Discussion

A key deployment question is whether a lightweight screening layer can detect adversarial inputs while maintaining a stable false-alarm rate. Our results show that feature-squeezing screening is most effective when inconsistency is measured in logit space rather than softmax space. On CIFAR-10, the clean-calibrated 5% FPR threshold transfers well to held-out clean data while preserving strong detection, supporting its practical use as an operational screening control [11].

These findings are consistent with prior wrapper-based detection approaches that probe output instability under benign transformations without retraining the protected model [2], [7], [8]. However, our results show that the scoring space is critical. Compared with softmax-based measures, logits-$\ell 2$ provides stronger separability and more consistent behaviour across attack objectives. With this choice, median filtering becomes a stable primary detector for both untargeted and targeted FGSM, and detection improves predictably as ϵ increases. This supports the view that a deployable screening control must not only be accurate but also predictable at a fixed operating point.

The comparison with a MagNet-style reconstruction benchmark further shows that reconstruction-based screening is competitive under FGSM, but the proposed median + logits-$\ell 2$ detector remains more stable under PGD-20. This aligns with prior work based on output divergence and uncertainty signals [5], [9], while preserving a lightweight wrapper-based design [2], [7]. Additional results on DenseNet121 and EfficientNet-B0 suggest that the signal is not restricted to a single backbone, although its strength remains classifier-dependent. At the same time, the extended evaluation reveals important limitations. Bit-depth reduction provides little useful separability, while NLMeans is more computationally expensive and weaker in some transfer-based black-box settings. The detector is also sensitive to certain benign corruptions: additive Gaussian noise causes a large increase in false positives, suggesting that high-frequency distortions can mimic adversarial instability. Robustness is also attack-dependent. The untargeted BPDA-lite approximation remains almost fully detectable, but the targeted threshold-aware adaptive attack preserves high attack success while sharply reducing recall. Transfer detection is asymmetric across source-target model pairs, and TinyImageNet results show weaker and more attack-dependent behaviour on a more complex dataset.

Overall, median + logits-$\ell 2$ offers the best balance between detection strength and runtime cost among the tested detectors. It is therefore best interpreted as a practical screening control and alert-triage mechanism, rather than as a general robustness guarantee. The method is strongest in the main CIFAR-10 setting, but can fail when adaptive optimisation suppresses detector scores, transfer perturbations fall near the threshold, or benign high-frequency corruptions resemble adversarial patterns. Therefore, the proposed method should be

interpreted within the evaluated image-classification settings, rather than as a broadly applicable adversarial defence across datasets, architectures, and adaptive threat models.

6. Conclusion

This work evaluated a lightweight, wrapper-based adversarial input screening approach based on feature squeezing and output inconsistency, operating at a fixed 5% FPR. On CIFAR-10, the results show that the classical softmax-$\ell 1$ inconsistency score provides limited detection, whereas logits-$\ell 2$ yields a much stronger and more stable screening signal. In particular, median + logits-$\ell 2$ is the strongest configuration among the tested squeezing-based detectors and remains more stable than a MagNet-style reconstruction baseline under the stronger PGD-20 setting. Additional transfer, distortion, architecture, and runtime analyses further support its practical value as a deployment-oriented input-screening and alert-triage mechanism in the evaluated image-classification settings. In addition, the analysis identifies certain weaknesses: The detector is less robust on TinyImageNet and shows reduced detection performance under more challenging attacks; transfer-based detection is shown to be asymmetric; and a targeted threshold-aware adaptive attack can sharply reduce recall while preserving high attack success. Thus, the results confirm the use of median + logits-$\ell 2$ as a practical screening control, but not as a general defence against fully adaptive adversaries.

Acknowledgement

The authors express their gratitude to the University of Vlora “Ismail Qemali” in Albania for its institutional support.

References

[1] Ren, Kui, Tianhang Zheng, Zhan Qin, and Xue Liu. "Adversarial Attacks and Defenses In Deep Learning." Engineering 6, no. 3 (2020): 346-360.

[2] Huang, Bo, Yi Wang, and Wei Wang. "Model-Agnostic Adversarial Detection by Random Perturbations." In IJCAI, pp. 4689-4696. 2019.

[3] Sharif, Mahmood, Sruti Bhagavatula, Lujo Bauer, and Michael K. Reiter. "Accessorize to A Crime: Real and Stealthy Attacks on State-Of-The-Art Face Recognition." In Proceedings of the 2016 acm sigsac conference on computer and communications security, pp. 1528-1540. 2016.

[4] Eykholt, Kevin, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. "Robust Physical-World Attacks on Deep Learning Visual Classification." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1625-1634. 2018.

[5] Deng, Zhijie, Xiao Yang, Shizhen Xu, Hang Su, and Jun Zhu. "Libre: A Practical Bayesian Approach to Adversarial Detection." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 972-982. 2021.

[6] Dong, Yinpeng, Chang Liu, Wenzhao Xiang, Hang Su, and Jun Zhu. "Competition on Robust Deep Learning." National Science Review 10, no. 6 (2023): nwad087.

[7] Xu, Weilin, David Evans, and Yanjun Qi. "Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks." arXiv preprint arXiv:1704.01155 (2017).

[8] Tian, Jinyu, Jiantao Zhou, Yuanman Li, and Jia Duan. "Detecting Adversarial Examples from Sensitivity Inconsistency of Spatial-Transform Domain." In Proceedings of the AAAI conference on artificial intelligence, vol. 35, no. 11, pp. 9877-9885. 2021.

[9] Aldahdooh, Ahmed, Wassim Hamidouche, Sid Ahmed Fezza, and Olivier Déforges. "Adversarial Example Detection for DNN Models: A Review and Experimental Comparison." Artificial Intelligence Review 55, no. 6 (2022): 4403-4462.

[10] Kumar, Siddheshwar, Shashank Srivastava, and Shashwati Banerjea. "Fortifying Vision Models: A Comprehensive Survey of Defences Against Adversarial Examples." Applied Soft Computing (2025): 113874.

[11] Vassilev, Apostol, Alina Oprea, Alie Fordyce, and Hyrum Andersen. "Adversarial Machine Learning: A Taxonomy and Terminology of Attacks And Mitigations." (2024).

[12] Macas, Mayra, Chunming Wu, and Walter Fuertes. "Adversarial Examples: A Survey of Attacks and Defenses in Deep Learning-Enabled Cybersecurity Systems." Expert Systems with Applications 238 (2024): 122223.

[13] Gong, Yuxin, Shen Wang, Xunzhi Jiang, Liyao Yin, and Fanghui Sun. "Adversarial Example Detection Using Semantic Graph Matching." Applied Soft Computing 141 (2023): 110317.

[14] Priya, Ch. E. N. Sai, and Manas Kumar Yogi. "Trustworthy AI Principles to Face Adversarial Machine Learning: A Novel Study." Journal of Artificial Intelligence and Capsule Networks 5, no. 3 (2023): 227-245. https://doi.org/10.36548/jaicn.2023.3.002.

[15] Nartker, Makaela, Zhenglong Zhou, and Chaz Firestone. "When Will AI Misclassify? Intuiting Failures on Natural Images." Journal of vision 23, no. 4 (2023): 4-4.

[16] Sitawarin, Chawin, Arvind Sridhar, and David Wagner. "Improving the Accuracy-Robustness Trade-Off for Dual-Domain Adversarial Training." In ICML 2021 Workshop on Uncertainty and Robust-ness in Deep Learning, vol. 3. 2021.

[17] Madry, Aleksander, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. "Towards Deep Learning Models Resistant to Adversarial Attacks." arXiv preprint arXiv:1706.06083 (2017).

[18] Tack, Jihoon, Sihyun Yu, Jongheon Jeong, Minseon Kim, Sung Ju Hwang, and Jinwoo Shin. "Consistency Regularization for Adversarial Robustness." In Proceedings of the AAAI conference on artificial intelligence, vol. 36, no. 8, pp. 8414-8422. 2022.

[19] Zhao, Shiji, Xizhe Wang, and Xingxing Wei. "Mitigating Accuracy-Robustness Trade-Off via Balanced Multi-Teacher Adversarial Distillation." IEEE transactions on pattern analysis and machine intelligence 46, no. 12 (2024): 9338-9352.

[20] Koenig, Matthias, Annelot W. Bosman, Holger H. Hoos, and Jan N. Van Rijn. "Critically Assessing the State of the Art in Neural Network Verification." Journal of Machine Learning Research 25, no. 12 (2024): 1-53.

[21] B., Shuriya, Kowsalya S., Varatharajan N., and Sivaraju S.S. 2025. “FLIDS: Fuzzy Logic-Based Framework for Interpretable Image Manipulation Detection”. Journal of Trends in Computer Science and Smart Technology 7 (3): 312-330. https://doi.org/10.36548/jtcsst.2025.3.002.

[22] Meng, Dongyu, and Hao Chen. "Magnet: A Two-Pronged Defense Against Adversarial Examples." In Proceedings of the 2017 ACM SIGSAC conference on computer and communications security,135-147. 2017.

[23] Carlini, Nicholas, and David Wagner. "Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods." In Proceedings of the 10th ACM workshop on artificial intelligence and security,3-14. 2017.

[24] Krizhevsky, Alex, Vinod Nair, and Geoffrey Hinton. "CIFAR-10 (Python version)." Data set. University of Toronto, 2009. https://www.cs.toronto.edu/~kriz/cifar.html.