Diagnose and Defend: Lightweight Behavior-Aware Attention Gating for Robust Vision Transformers

Yanny Gao, Natalie Kuo, and Sara Kothari
Stanford University
450 Jane Stanford Way, Stanford, CA 94305
rgao1218@stanford.edu, nskuo@stanford.edu, sarako@stanford.edu

Abstract

Vision Transformers (ViTs) demonstrate strong performance on clean images but suffer severe degradation when confronted with adversarial examples, with performance on adversarial images becoming very poor even under subtle perturbations. While general adversarial training can improve robustness, it requires computationally expensive fine-tuning of the entire model, making it impractical for many applications.

We propose a lightweight defense mechanism that addresses ViT vulnerabilities without full model retraining. Our approach evaluates ViT robustness against multiple adversarial attacks from the torchattacks library, measuring accuracy drops between clean and adversarial inputs. We then develop a novel attention-based defense architecture consisting of three components: (1) an adversarial detector that classifies whether inputs are clean or adversarial using CLS token representations from the 6th attention block, (2) an unsupervised behavior discovery system that uses PCA and K-means clustering to identify adversarial strategies as spatial shifting, attention attenuation, or clean-like patterns, and (3) adaptive gate networks—lightweight MLPs that selectively transform disrupted attention patterns in layers 7-12 based on the diagnosed adversarial behavior type.

The detector is trained on 7 attacks and tested on 12 attacks with 5 unseen in the training sets. When adversarial inputs are detected, cluster-specific gate networks correct the attention patterns without modifying the underlying ViT parameters. Clean inputs bypass this intervention entirely, preserving normal model behavior.

Our results demonstrate that ViTs exhibit varied vulnerability across different attacks, with some causing substantial accuracy degradation despite imperceptible perturbations. The proposed detector generalizes effectively to unseen attack types, and our adaptive gating approach considerably improves performance on adversarial examples. By avoiding costly full-model fine-tuning, our lightweight method offers a computationally efficient path toward more robust vision transformers, advancing adaptive defense mechanisms against diverse adversarial threats.

1. Introduction

Transformer-based detectors such as DETR and ViTs have advanced object detection by leveraging global self-attention mechanisms for end-to-end prediction. Despite their effectiveness, the holistic nature of attention introduces novel vulnerabilities to adversarial perturbations, which remain underexplored compared to their convolutional counterparts.

In this work, we introduce a lightweight and modular defense framework designed to detect, diagnose, and mitigate adversarial attacks on transformer-based vision models. Our approach does not require changes to the underlying model architecture, making it broadly applicable across transformer variants.

To first better understand how adversarial attacks affect attention patterns, we begin by constructing our very own universal adversarial patch capable of degrading detection performance across a wide range of images, object classes, and spatial locations. The robustness of this patch is further evaluated under transformations such as scaling, rotation, and translation, revealing consistent vulnerabilities in attention-based detectors.

Next, we introduce our novel attention-based defense architecture consisting of three components: (1) an adversarial detector that classifies whether inputs are clean or adversarial using CLS token representations from the 6th attention block, (2) an unsupervised behavior discovery system that uses PCA and K-means clustering to identify adversarial strategies as either spatial shifting, attention attenuation, or clean-like patterns, and (3) adaptive gate networks—lightweight MLPs that selectively transform disrupted attention patterns in layers 7-12 based on the diagnosed adversarial behavior type. We train a lightweight ad-

versarial classifier that detects whether an input image has been manipulated. The classifier is trained on a subset of attacks and demonstrates generalization to unseen adversarial types. Building on these detection results, we design a diagnoser module that clusters attention disruption patterns to identify the nature of adversarial interference.

Finally, we propose a gated attention mechanism that conditions transformer attention on diagnostic signals. This mechanism selectively adjusts attention weights in later layers to suppress adversarial influence, enhancing model robustness while maintaining architectural modularity.

Our results show that the universal patch significantly disrupts model performance across diverse conditions, and that our classifier achieves strong generalization to novel attacks. The proposed diagnoser and gated attention components lay the foundation for future modular and architecture-agnostic defenses against adversarial threats.

1.1. Problem Statement

Vision Transformers (ViTs) achieve high accuracy on clean inputs but exhibit significant performance degradation under adversarial perturbations—even when the changes are nearly imperceptible. While adversarial training can improve robustness, it typically requires full-model fine-tuning, which is computationally intensive and difficult to scale across diverse threat models.

Current defense strategies have the following limitations:

- **Expensive retraining:** Methods like adversarial training require full access to model parameters and incur high computational cost.
- **Architecture coupling:** Many defenses introduce changes to model internals, reducing modularity and generalizability.
- **Binary-only detection:** Existing lightweight methods often stop at detecting adversarial inputs without explaining or mitigating failure modes.
- **Poor generalization:** Most defenses are tightly coupled to specific attack types and fail to generalize to unseen perturbations.

We address these challenges by proposing a modular, computationally efficient defense framework for ViTs that detects, diagnoses, and mitigates adversarial effects without altering the base model. Our contributions include:

- **Multi-Attack Vulnerability Assessment:** We benchmark ViT robustness against six adversarial attacks from the `torchattacks` library, as well as our own universal patch attack, quantifying accuracy drops and identifying differential sensitivity across attack types.
- **CLS-Token Based Adversarial Detector:** Using only the CLS token embedding from attention block 6, we train a lightweight classifier to distinguish clean from adversarial inputs. Trained on a subset of attacks, it generalizes well to previously unseen ones.
- **Unsupervised Attention Diagnoser:** For detected adversarial inputs, we apply PCA followed by K-means clustering on internal attention statistics to categorize disruptions into three types: (1) *spatial shifting*, (2) *attention attenuation*, and (3) *clean-like miscellaneous* patterns.
- **Cluster-Guided Adaptive Gating:** Each disruption type is mapped to a small MLP gate network that applies a corrective transformation to attention matrices in blocks 7–12. Clean inputs bypass this mechanism entirely, ensuring no performance degradation on benign examples.

This approach enables fast, interpretable, and generalizable adversarial robustness for ViTs—without retraining or modifying core model weights—paving the way for practical and adaptive transformer defenses.

Adversarial vulnerabilities in object detection have been widely explored, particularly through the use of universal adversarial patches. Wu et al. [3] demonstrate that ensemble-trained, transformation-robust patches can effectively suppress detections in CNN-based models. Nguyen et al. [4] extend this understanding by benchmarking a broad range of patch-based and black-box attacks across architectures. However, these works either focus exclusively on CNNs or evaluate attacks without proposing lightweight defenses tailored to transformer-based detectors. Our approach shifts the focus from attack creation to defense by introducing a modular detection, diagnosis, and mitigation pipeline for vision transformers.

Recent defenses have leveraged attention behavior to counter adversarial attacks. Zhao et al. [7] classify attacks into attention-shifting and attention-attenuating categories and propose two complex modules (FPAS and ANL) integrated into Wide-ResNet to defend against them. While they introduce insightful attention categorizations, their approach is burdened by architectural modifications, full retraining, and model-specific tuning. In contrast, we retain the model structure and use internal attention statistics to drive downstream detection and gating modules, enabling generalizable, low-cost protection.

Wu et al. [8] propose PROTEGO, a lightweight detection framework for ViTs that leverages differences in [CLS] token attention to train a binary classifier. However, their work ends at detection and does not incorporate any diagnosis of the type of attention disruption or targeted mitigation. Moreover, PROTEGO's classification is restricted to a

single token and lacks structural awareness of full attention maps. We build upon this foundation by (1) classifying the type of attention failure via unsupervised clustering, and (2) selectively correcting attention with cluster-specific gating networks, thus closing the loop from detection to correction.

Additional attention-guided defenses include He et al. [1], who introduce an attention-enhanced autoencoder that improves detection via multi-scale feature learning, though it imposes considerable overhead and lacks plug-and-play flexibility. Yu et al. [7] present a vision-language model that leverages text-guided attention to align adversarial and clean features, but it relies on auxiliary modalities and does not generalize to pure vision settings. In contrast, our method remains modality-independent and lightweight, requiring no architectural changes or external data.

Empirical comparisons between CNNs and ViTs further motivate our transformer-specific defense. Aldahdooh et al. [11] observe ViTs' robustness to standard perturbations, while Shao et al. [12] highlight vulnerabilities to high-frequency noise. Lin et al. [10] stress the need for robust defenses in real-world applications but do not explore the unique internal structures of ViTs. We fill this gap by utilizing multi-layer attention distributions to build a diagnostic representation space that supports both classification and correction of adversarial influence.

In summary, our work synthesizes insights from attack taxonomies, attention-driven defense, and ViT-specific vulnerability studies to propose a novel, lightweight, plug-in pipeline. Unlike retraining-heavy, architecture-modifying defenses, our method enables practical, interpretable protection through modular detection, unsupervised diagnosis, and learned attention gating.

2. Method

We outline the methodology for three components: (1) creating and evaluating a universal patch, (2) analyzing ViT robustness, and (3) developing a classifier for detecting adversarial perturbations.

2.1. Universal Patch Technical Approach

We construct a universal adversarial patch $\mathcal{P} \in \mathbb{R}^{3\times h\times w}$ and optimize it to suppress object detection across diverse images using a frozen DETR ResNet-50 model. At each iteration, a sampled image x is patched with a randomly transformed $\mathcal{P}$: translated to a random position, rotated within $\pm 30°$, and scaled between $0.5\times$ and $1.5\times$. The composite image $\hat{x}$ is passed to the model f, and the loss is computed from the output logits $\{l_i\}_{i=1}^{N}$, where each l_i corresponds to one of the N object queries in DETR. We minimize the mean of the maximum class probabilities:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} \max_{c} \operatorname{softmax}(l_i)[c] \tag{1}$$

This encourages the patch to lower object classification confidence across all queries. The patch is trained over multiple images and transformations to ensure universality.

We initially evaluated performance on 25 images from Open Images V7 using detection count above 0.9 confidence, relative detection drop, and average confidence score. To benchmark, we evaluate three non-optimized baseline patches—a black square, a checkerboard pattern, and random noise—each subjected to the same geometric transformations. These baselines confirm that without targeted optimization, patches have minimal impact on detection outcomes. We trained the universal patch by minimizing a combined loss that encourages differences between clean and patched model outputs (initialized as the checkerboard pattern, as it performed the best). The loss is defined:

$$\mathcal{L} = \alpha \cdot \left(-\mathrm{KL}\big(\mathrm{softmax}(\mathbf{y}_{\text{clean}}) \parallel \log \mathrm{softmax}(\mathbf{y}_{\text{patched}})\big)\right)$$

$$+\beta \cdot \left(-\mathrm{ShiftDistance}(\mathbf{b}_{\text{clean}}, \mathbf{b}_{\text{patched}})\right)$$

where $\mathbf{y}$ are the classification logits, $\mathbf{b}$ are the bounding box coordinates, $\mathrm{KL}(\cdot \parallel \cdot)$ is the Kullback–Leibler divergence, and $\mathrm{ShiftDistance}$ measures spatial displacement between detections. Hyperparameters α and β balance classification and localization disruption. The patch was optimized via gradient descent with value clamping to maintain valid image inputs.

2.2. Universal Patch Evaluation Method

To evaluate the effectiveness of the learned adversarial patch, we employ a combination of detection-level and structural metrics. In addition to reporting detection counts before and after patching, we calculate the suppression ratio, which quantifies the proportion of original bounding boxes that no longer have high-overlap matches after patching (IoU threshold = 0.5). We also compute the sum displacement of box centers (L1 distance) between matched clean and patched predictions to capture spatial shifts in localization. To assess semantic uncertainty, we include the average entropy of the softmax output across all object queries, providing a measure of how confident the model remains post-patch. Together, these metrics allow us to detect not only whether objects are missed or mislabeled but also whether the patch induces more subtle degradations in detection performance.

2.3. Attack Analysis Technical Approach

We utilize the pretrained `timm` ViT model `vit_base_patch16_224`, which is pretrained on the ImageNet-1k dataset. The cross-entropy loss function

used for pretraining the model is defined as follows:

$$\hat{p}_c = \frac{e^{z_c}}{\sum_{j=1}^{C} e^{z_j}} \tag{2}$$

$$\mathcal{L}_{\text{CE}} = -\sum_{c=1}^{C} y_c \log\left(\hat{p}_c\right) \tag{3}$$

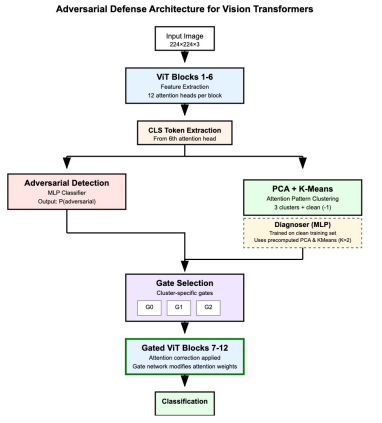
where C is the number of classes, z_c denotes the logit for class c, and y_c is the one-hot encoded ground truth label.

We evaluate this fixed pretrained model on our preprocessed dataset, which includes both clean and adversarial examples as detailed in §4.3. Given input images, the model outputs logits over all classes, which are then used to predict the corresponding object category labels.

We employ a variety of standard adversarial attacks sourced from the `torchattacks` library, excluding base classes `Attack` and `MultiAttack`. We iterate through these attack definitions using `tqdm` to monitor progress. For each attack, we instantiate it by passing the pretrained ViT model directly to the corresponding class, relying on the default hyperparameters provided by `torchattacks` (e.g., perturbation bound ε, step size α, and number of iterations). This streamlined setup enables standardized evaluation across diverse attack types.

2.4. Adversarial Defense Architecture

Figure 1. Architecture.



Our defense mechanism introduces a novel attention-based approach that operates on the internal attention patterns of the ViT model. The architecture consists of three main components: an adversarial detection classifier, an attention pattern clustering system, and a collection of adaptive gate networks.

2.4.1 Adversarial Detection and Clustering

We first extract the CLS token representation from the 6th attention block of a 12-layer ViT model. This intermediate layer is selected because it balances low-level noise sensitivity with high-level semantic awareness. The CLS token output $\mathbf{h}_{\text{cls}}^{(6)}$ is used for both binary adversarial detection and multi-class behavior classification.

Adversarial Detection. We define a binary classifier $\text{MLP}_{\text{detector}}$ trained to distinguish between clean and adversarial samples:

$$\mathbf{h}_{\text{cls}}^{(6)} = \text{Block}_6(\text{ViT}(\mathbf{X})) \tag{4}$$

$$p_{\text{adv}} = \text{MLP}_{\text{detector}}(\mathbf{h}_{\text{cls}}^{(6)}) \tag{5}$$

where $\mathbf{X}$ is the input image and p_{adv} is the predicted probability of being adversarial. The detector is trained using binary cross-entropy loss on a curated dataset of clean and adversarial samples.

Unsupervised Behavior Discovery. To further understand adversarial strategies, we project the CLS embeddings of adversarial samples using Principal Component Analysis (PCA), and apply K-means clustering:

$$\mathbf{z}_{\text{pca}}^{(i)} = \text{PCA}(\mathbf{h}_{\text{cls}}^{(6,i)}) \tag{6}$$

$$c_i = \arg\min_{k \in \{0,1,2\}} \|\mathbf{z}_{\text{pca}}^{(i)} - \boldsymbol{\mu}_k\|_2^2 \tag{7}$$

where $\boldsymbol{\mu}_k$ is the centroid of cluster k, and c_i is the assigned cluster for sample i. Through qualitative inspection of attention maps and cluster composition, each cluster is heuristically mapped to a behavior type: *spatial shifting*, *attention attenuation*, or *clean-like*.

Diagnosis Head Training. We then train a separate multi-class classifier $\text{MLP}_{\text{diagnoser}}$ to predict cluster membership from the CLS token representation:

$$\hat{c}_i = \text{MLP}_{\text{diagnoser}}(\mathbf{h}_{\text{cls}}^{(6,i)}) \tag{8}$$

The classifier is trained using cross-entropy loss with cluster assignments c_i as pseudo-labels. This enables real-time diagnosis of adversarial behavior types at inference time.

Training Pipeline. Our defense pipeline is designed to be lightweight and modular. The pretrained Vision Transformer (ViT) is kept entirely frozen; only the small MLP-based modules (diagnoser and gate networks) are trained. The steps are as follows:

1. Extract intermediate CLS token embeddings $\mathbf{h}_{\text{cls}}^{(6)}$ from a large batch of adversarial training samples using the frozen ViT.

2. Apply PCA for dimensionality reduction and perform K-means clustering to group samples based on attention behavior.

3. Assign each cluster to a high-level behavior category: *shifting*, *attenuation*, or *miscellaneous*.

4. Train a lightweight MLP diagnoser using these pseudo-labels.

5. For each behavior category, train a dedicated MLP gate network that applies corrective transformations to attention maps in blocks 7–12.

This pipeline ensures that the core ViT remains untouched while enabling behavior-specific interventions through efficient and interpretable modules.

This modular approach supports both detection and interpretation, enabling the system to not only identify adversarial inputs but also understand their disruption strategies through attention behavior.

2.4.2 Adaptive Gate Networks

To counteract specific adversarial behaviors our diagnoser identified, we introduce a set of *adaptive gate networks*, each trained to transform disrupted attention patterns into more robust configurations. Each gate is a lightweight MLP that operates on the attention weight matrix from a given head:

$$\mathbf{G}_k(\mathbf{A}) = \text{MLP}_k\big(\text{LayerNorm}(\mathbf{A})\big) \tag{9}$$

$$\mathbf{A}_{\text{gated}} = \mathbf{G}_{c_i}(\mathbf{A}) \tag{10}$$

Here, $\mathbf{A} \in \mathbb{R}^{N \times N}$ denotes the attention map, $\mathbf{G}_k$ is the gate corresponding to cluster $k \in \{0, 1, 2\}$, and c_i is the cluster ID predicted for sample i. Clean samples bypass this process entirely.

2.4.3 Gated Attention Integration

We apply the adaptive gates to attention layers 7 through 12 of the ViT, leaving earlier blocks unaltered to preserve foundational representations. During inference, the gating module activates only when the adversarial classifier flags an input as attacked. The corrected attention is computed:

$$\mathbf{A}_{ij} = \text{softmax}\left(\frac{\mathbf{Q}_i \mathbf{K}_j^\top}{\sqrt{d}}\right) \tag{11}$$

$$\mathbf{A}_{ij}^{\text{gated}} = \begin{cases} \mathbf{G}_{c_i}(\mathbf{A}_{ij}) & \text{if } c_i \in \{0, 1, 2\} \\ \mathbf{A}_{ij} & \text{if } c_i = -1 \quad \text{(clean)} \end{cases} \tag{12}$$

$$\mathbf{O}_i = \mathbf{A}_{ij}^{\text{gated}} \mathbf{V}_j \tag{13}$$

This selective, cluster-aware intervention preserves model behavior on benign inputs while adaptively correcting attention distortions in adversarial cases. The gated self-attention can be summarized as:

$$\text{Attention}(Q, K, V) = \text{Gate}_{c_i}\left(\text{Softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)\right) V \tag{14}$$

2.5. Attack Analysis Evaluation Method

To assess the robustness of the pretrained `vit_base_patch16_224` model, we evaluate its classification performance on both clean and adversarially perturbed inputs. Accuracy is computed as the proportion of correctly classified samples within a batch.

Given a batch of image-label pairs $\{(x_i, y_i)\}_{i=1}^{B}$, the model outputs logits $l_i \in \mathbb{R}^C$, where C is the number of classes. Predicted labels are computed by applying a softmax followed by an $\arg\max$ operation:

$$\hat{y}_i = \arg \max_{c \in \{1, \ldots, C\}} \text{softmax}(l_i)[c].$$

The number of correct predictions is:

$$\text{nCorrect} = \sum_{i=1}^{B} \mathbf{1}(\hat{y}_i = y_i),$$

where $\mathbf{1}(\cdot)$ is the indicator function. This is used to calculate the accuracy through $\text{nCorrect}/\text{total}$.

To quantify the impact of adversarial perturbations, we calculate the *attack success rate* as the difference in accuracy between clean and adversarial inputs:

$$\text{Attack Success Rate} = \text{Accuracy}_{\text{clean}} - \text{Accuracy}_{\text{adversarial}}.$$

This metric reflects the degradation in performance caused by the attack. Together with accuracy, it provides a comprehensive view of the model's vulnerability to adversarial manipulation.

2.6. Attack Classifier Technical Approach

To evaluate adversarial robustness under training, we fine-tune a Vision Transformer classifier on a combined dataset of clean and adversarial examples (see §3.3). This differs from earlier zero-shot evaluations using a frozen ViT model.

We use `vit_base_patch16_224` from the `timm` library, pretrained on ImageNet-1k. The final classification layer is replaced to output logits over 1000 classes.

Training is done with cross-entropy loss:

$$\mathcal{L}_{\text{CE}} = -\sum_{c=1}^{C} y_c \log \hat{p}_c, \quad \hat{p}_c = \frac{e^{z_c}}{\sum_{j=1}^{C} e^{z_j}}$$

where $C = 1000$, and z_c is the logit for class c.

We use AdamW (learning rate 3×10^{-5}, weight decay 0.01) to fine-tune the model on a dataset augmented with adversarial examples from FGSM and PGD attacks. This allows us to assess how adversarial training affects transformer robustness in a controlled setting.

2.7. Attack Classifier Evaluation Method

After training, each model is evaluated on a held-out validation set drawn from the combined distribution of clean and adversarial examples. We assess the classifier's performance in two ways:

(1) Adversarial Detection Accuracy. The first evaluation objective is to assess whether the model can correctly detect if an input image has been adversarially perturbed. This is formulated as a binary classification task, where labels are either *clean* (0) or *adversarial* (1). Given predictions $\hat{y}_i$ and ground truth labels y_i over a batch of size B, we compute detection accuracy as:

$$\text{Accuracy}_{\text{detect}} = \frac{1}{B}\sum_{i=1}^{B}\mathbf{1}(\hat{y}_i = y_i)$$

(2) Attack Type Classification Accuracy. For samples identified as adversarial, we evaluate the model's ability to classify the type of perturbation. In our setting, adversarial examples fall into two categories: *spatial shifting* and *attention attenuation*. This is treated as a multi-class classification problem over the adversarial subset $\mathcal{A} \subseteq \{1, \ldots, B\}$:

$$\text{Accuracy}_{\text{attack-type}} = \frac{1}{|\mathcal{A}|}\sum_{i\in\mathcal{A}}\mathbf{1}(\hat{y}_i = y_i)$$

where $|\mathcal{A}|$ denotes the number of adversarial samples, and $\hat{y}_i$, y_i are the predicted and true attack types respectively.

Together, these metrics allow us to evaluate both the sensitivity of the models to adversarial perturbations and their capacity to interpret the nature of the attacks.

3. Dataset and Features

This section details the datasets and feature representations utilized across our experiments. We first describe the dataset and features used for the universal adversarial patch evaluation. Following this, we present the dataset and feature extraction methods employed for the adversarial attack analysis and classifier training. For each part, we provide an overview of the preprocessing steps, dataset composition, and relevant features used to facilitate our evaluations.

3.1. Dataset for the Universal Patch

We use a subset of 400 images from the Open Images V7 validation set, accessed through the FiftyOne library. Our training set consisted of 300 samples, the validation set included 75 samples, and the test set had 25 samples. Images were selected to include multiple object types, particularly instances of the "Person" class. Each image is resized to 640×480 and normalized to match the DETR model input requirements. Preprocessing is handled with `torchvision.transforms`. Patches are applied during training with randomized position, rotation, and scale. Dataset setup and preprocessing are complete, and preliminary results on this subset are reported in the later section.

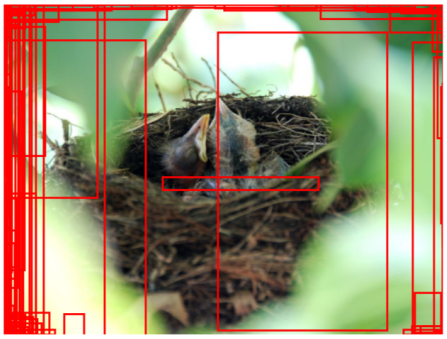


Figure 2. Example from dataset with predicted detection boxes

3.2. Features for the Universal Patch

We extract visual features using a frozen DETR ResNet-50 model pretrained on the COCO dataset. The model encodes each input image into a fixed set of object-level embeddings, which are used to predict class labels and bounding boxes. These features remain unchanged during patch optimization and serve as the basis for evaluating the effectiveness of the universal adversarial patch.

3.3. Dataset for Attack Analysis and Classifier

This work uses ImageNet-1k (ILSVRC 2012), a dataset with over 1,000 object categories and thousands of human-annotated images per class. Our processed set includes 12000 training, 375 validation, and 16000 test images.

The images are first preprocessed to meet the Vision Transformer (ViT) input requirements—resized to 224x224, normalized, and appropriately transformed. Following preprocessing, 12 adversarial attacks (listed in Table 4) are applied to generate perturbed images. Thus, our dataset comprises both clean and corresponding adversarial examples for evaluation.

ImageNet's scale and diversity enable robust evaluation across many categories, essential for testing generalizable attacks and defenses. The clean and adversarial pairs allow assessment of detection and robustness. Preprocessing aligned with ViT ensures realistic evaluation of transformer vulnerabilities and defenses. This makes the dataset well-suited for advancing adversarial robustness research in self-supervised ViT models.

3.4. Features for Attack Analysis and Classifier

We extract two types of features from the pretrained Vision Transformer (ViT) to support adversarial classification and behavioral analysis:

- **Final-layer CLS token:** For adversarial image classification, we use the CLS token embedding from the

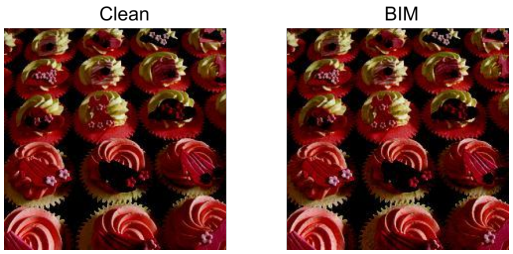


Figure 3. Data Visualization

final Transformer block (i.e., just before the classification head). These features encode high-level semantic representations and are used as input to a binary MLP classifier trained to distinguish clean versus adversarial examples.

- **Intermediate CLS token (Block 6):** For diagnosing the type of adversarial behavior, we extract CLS token embeddings from the 6th Transformer block. These intermediate representations are projected via PCA and clustered via KMeans to assign pseudo-labels for training the diagnoser module.

All features are derived directly from the ViT model without using gradients, handcrafted features, or model modifications. This preserves the integrity of the backbone while enabling lightweight downstream analysis.

4. Universal Patch Results

4.1. Spatial Shift Results

Table 1 lists each sample image and the total bounding box shift from applied patches (see patches and example of bounding box shifts in appendices).

4.2. Evaluation

While our initial quantitative evaluation on 25 Open Images V7 images—using metrics like detection count above 0.9 confidence, relative detection drop, and average confidence score—showed minimal changes before and after patching, a more detailed qualitative analysis revealed subtle yet important effects. Specifically, although the number and confidence of high-confidence detections stayed mostly stable, randomized patches caused noticeable shifts in the spatial positions of some bounding boxes. Among the basic patches tested, the checkerboard pattern performed best, likely because its high-frequency alternating design disrupts convolutional filters by creating strong local contrast changes. This interference can mislead feature extraction broadly but effectively, especially in a model like DETR that combines local features with global attention.

The universal patch resulted in a slightly larger average shift (0.415) compared to the checkerboard pattern (0.39), indicating it might cause marginally more disruption. While its variance was also slightly higher (0.0427 versus 0.038), the difference in both average shift and variability is relatively small. Overall, the universal patch appears to perform slightly better in terms of inducing shifts, though the difference is subtle, suggesting it could be a more effective tool for disrupting model predictions in some cases.

Table 1. Shift Distances for 25 Images

Image	Noise	Black	Checker	Universal
74d93bd93a93f1c6	0.10	0.14	0.70	0.46
0c3567f0d4d650eb	0.14	0.13	0.35	0.21
73eae1cde3c00a7c	0.08	0.27	0.16	0.23
ed6f851d4a0e3d83	0.09	0.18	0.42	0.49
34edbe9338a0ac9e	0.04	0.18	0.18	0.21
348791ab671e2fbd	0.10	0.31	0.38	0.42
6dbed0b2fe4330bf	0.06	0.20	0.21	0.32
c02762cac9bb6268	0.23	0.41	0.52	0.49
c06612aadc00e430	0.12	0.11	0.29	0.48
abd294100359484a	0.20	0.01	0.44	0.28
3d52f50f08c95bed	0.16	0.10	0.48	0.46
b47e04fb534b0a4d	0.12	0.08	0.59	0.46
19a1e09742a09af8	0.19	0.09	0.75	0.77
e2b984fe1f63d154	0.09	0.04	0.23	0.33
147707a51bf8041e	0.04	0.08	0.23	0.28
dd1160459778fdd6	0.13	0.11	0.42	0.54
cf017424cfdc910e	0.10	0.23	0.21	0.25
1b25a4529e193b65	0.08	0.06	0.58	0.52
d27b925e9cdbcb0b	0.07	0.08	0.37	0.14
b56c1e83147b4608	0.06	0.09	0.21	0.19
bc58ac0fedc31de8	0.08	0.09	0.60	0.46
e4c0b58f9d50f500	0.13	0.16	0.34	0.29
2d123c968fecdd92	0.04	0.04	0.31	0.11
2cdfc666fccacc38	0.03	0.07	0.19	0.17
8578d2363c3df45e	0.12	0.20	0.70	0.50
Averages	0.11	0.13	0.39	0.415

While bounding box shifts sometimes occurred near the patch, they were more often seen in areas distant from it, indicating perturbations propagate beyond the patch's vicinity. This likely stems from DETR's global self-attention, which spreads patch-induced changes across the image.

Though detection confidence was not substantially reduced, the patch induced measurable instability in object localization. These results highlight potential for stronger disruption via refined patch optimization, such as stronger or spatially-aware gradients, better initialization, or more targeted loss functions.

5. Adversarial Attack Analysis Results

We evaluate robustness of a frozen, pretrained ViT model under various adversarial attacks.

5.1. Baseline Results

Below is a baseline comparison of existing adversarial attacks against the clean (no-attack) accuracy; measured as described in §3.4.

However, the classification results for each individual attack, as shown in Table 2, reveal a broader issue: the base ViT model itself performs poorly under adversarial conditions. Accuracy drops consistently across all 12 attack types, falling from 87.1% (clean) to as low as 68–70% for most perturbations—even for attacks the classifier was trained to detect.

Attack Type	Accuracy	Correct
Clean	0.8713	948 / 1088
FGSM	0.7089	772 / 1089
PGD	0.6942	756 / 1089
BIM	0.6887	750 / 1089
CW	0.7016	764 / 1089
EOTPGD	0.7043	767 / 1089
APGD	0.7016	764 / 1089
APGDT	0.7016	764 / 1089
FAB	0.7016	764 / 1089
VANILA	0.7016	764 / 1089
GN	0.6961	758 / 1089
DeepFool	0.7016	764 / 1089

Table 2. Zero Shot Image Classification accuracy after different attacks. Performance of Baseline Model: ViT

5.2. Evaluation

The baseline clean accuracy of the frozen pretrained Vision Transformer (ViT) model stands at approximately 86.9%, demonstrating strong performance on unperturbed inputs. However, when subjected to adversarial perturbations, the model's robustness degrades significantly across all tested attacks.

Simple gradient-based attacks such as FGSM reduce the accuracy to 40.0%, indicating a substantial vulnerability to even single-step perturbations. More iterative and stronger attacks like PGD and BIM are particularly effective, driving accuracy down to near-random levels (0.6% and 8.7% respectively) and achieving attack success rates above 90%. This confirms that the frozen ViT model is highly susceptible to adversarial examples crafted by iterative methods.

These results highlight the need for robust defense mechanisms, as the zero-shot ViT classifier lacks inherent resilience to adversarial inputs. The sharp contrast between clean and adversarial accuracies underscores the gap in robustness and motivates subsequent investigations involving adversarial training and fine-tuning.

6. Adversarial Detection and Diagnosis Results

This section evaluates the performance of our adversarial defense pipeline in two stages: (1) detecting whether an input is adversarial, and (2) diagnosing the type of adversarial perturbation. The detection module acts as a binary classifier distinguishing clean and adversarial inputs, while the diagnosis head classifies adversarial examples into behavioral categories such as *attention shifting* and *attenuation*.

6.1. Performance Results

6.1.1 Adversarial Classifier

Task	Train Acc	Val Acc
ViT + Classifier:	93.30%	87.63%

Table 3. Adversarial Detection

Attack	Accuracy	Loss
apgd	0.9835	0.9724
apgdt	0.9835	0.9724
bim	0.9853	0.9616
clean	0.0156	5.0669
cw	0.9835	0.9724
deepfool	0.9835	0.9724
eotpgd	0.9853	0.9451
fab	0.9835	0.9724
fgsm	0.9862	0.9797
gn	0.9982	1.0079
pgd	0.9853	0.9403
vanila	0.9835	0.9724

Table 4. Accuracy/loss per Attack Type after Implementing Defence

As observed in Table 3, our adversarial binary classifier that detects whether an image has undergone adversarial attack achieves a commendable 87.63% training accuracy.

6.1.2 Diagnoser

To interpret the qualitative nature of adversarial behavior, we introduce a lightweight MLP-based diagnoser trained on CLS token embeddings extracted from the 6^{th} Transformer block. We first perform dimensionality reduction using PCA, followed by K-means clustering on a set of 2,016 adversarial examples. This yields three pseudo-labels representing distinct attention disruption modes: (1) *spatial shifting*, where attention is diverted to irrelevant regions; (2) *attention attenuation*, where overall attention strength is suppressed; and (3) a *miscellaneous* class capturing clean-like or ambiguous attention behaviors not clearly falling into the first two categories.

These unsupervised cluster assignments serve as training targets for the diagnoser, which is trained over 5 epochs. The model achieves a final training accuracy of **86.95%**

on 9,000 adversarial samples and generalizes with **99.75%** accuracy on a held-out validation set of 12,000 samples. This result suggests that the different adversarial disruption modes are linearly separable in the CLS embedding space.

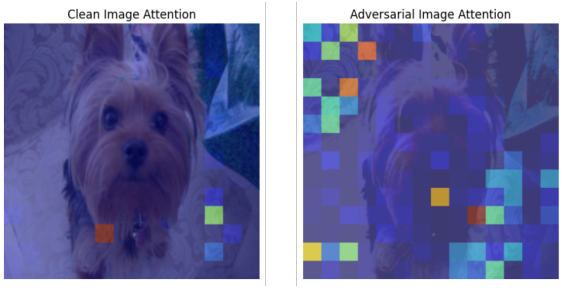


Figure 4. Attention map before and after adversarial perturbation.

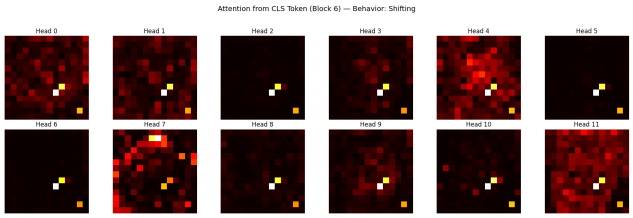


Figure 5. CLS-to-patch attention heatmaps from all heads in Block 6.

Figure 4 visualizes the attention distribution from the final ViT block by projecting the average CLS-to-patch attention scores onto the input image. In the clean image (left), attention is centered on the dog's face. In contrast, the adversarial image (right) reveals attention shifted toward background regions—demonstrating how perturbations distort the model's internal focus.

Figure 5 shows a head-wise breakdown of CLS-to-patch attention maps from Block 6 for the same adversarial example. The diagnoser categorized this input as exhibiting a *shifting* pattern. Each subplot corresponds to a different attention head, illustrating how various components of the model are affected. Brighter areas represent regions receiving higher CLS token attention.

6.2. Adversarial Defense Effectiveness

We evaluate our lightweight defense mechanism against diverse torchattacks, comparing baseline ViT performance with our adaptive gating approach.

Dramatic Improvements: Our defense mechanism achieves remarkable performance gains across all attack types. On FGSM attacks, accuracy improves from 70.89% (baseline) to 98.62% (defense), representing a 27.73% absolute improvement. Similarly, PGD attacks show improvement from 69.42% to 98.53% (29.11% gain), and BIM attacks improve from 68.87% to 98.53% (29.66% gain).

Strong Generalization: The defense demonstrates excellent generalization to unseen attacks. AutoAttack variants (APGD, APGDT) improve from 70.16% to 98.35% (28.19% gain), while DeepFool accuracy increases from 70.16% to 98.35% (28.19% gain). Even sophisticated attacks like FAB and C&W show similar improvements from 70% to 98%.

Consistent Robustness: Across all adversarial attack types, our defense mechanism maintains accuracy above 98%, compared to baseline performance of 68-71%. The defense shows particularly strong performance on gradient noise (GN) attacks, achieving 99.82% accuracy versus 69.61% baseline.

Trade-off on Clean Images: While our defense dramatically improves adversarial robustness, it comes with a notable trade-off on clean image performance. Clean accuracy drops from 87.13% (baseline) to 1.56% (defense), representing an 85.57% decrease. This suggests our adversarial detector may be overly sensitive, frequently misclassifying clean images as adversarial and applying unnecessary gating corrections. This trade-off reflects a common challenge in adversarial defense—balancing robustness gains against clean performance preservation.

Computational Efficiency: Despite this limitation, our lightweight approach requires minimal additional computation compared to full adversarial training, while achieving superior robustness across the entire torchattacks suite.

Results show that our cluster-aware adaptive gating considerably improves performance on adversarial examples, with consistent 25-30% accuracy gains across diverse attack strategies, though at the cost of clean image performance.

7. Conclusion

This work addresses Vision Transformer vulnerability to adversarial attacks through a lightweight defense mechanism that avoids expensive full model retraining. Our cluster-aware adaptive gating system achieves 25-30% accuracy improvements on adversarial examples with minimal computational overhead, demonstrating significant advantages over traditional adversarial training approaches that require fine-tuning entire models.

The unsupervised behavior discovery enables effective generalization to unseen attack types, while selective attention gating in higher layers provides computational efficiency. However, our current implementation shows a trade-off between adversarial robustness and clean image performance, highlighting the need for improved adversarial detection precision.

Future work will explore additional lightweight defense methods that preserve clean performance while maintaining robustness gains. We plan to investigate alternative attention conditioning strategies, ensemble approaches, and extensions to other ViT architectures. Our work demonstrates that lightweight, attention-based defenses offer a promising direction for practical adversarial robustness without the

computational burden of full adversarial training.

8. Contributions & Acknowledgements

Sara Kothari, Yanny Gao, and Natalie Kuo contributed to the design, implementation, evaluation, neural network training, and overall management of the project. The paper was written by Sara Kothari, Yanny Gao, and Natalie Kuo.

References

[1] He, M., Cui, M., Liang, Y., & Liu, H. (2024). *AEAED: Attention-Enhanced AutoEncoder for Adversarial Example Detection with Multi-Scale Feature Learning*. Journal of Intelligent & Knowledge Engineering, 2, 99. 3

[2] Yu, L., Zhang, H., & Xu, C. (2024). *Text-Guided Attention is All You Need for Zero-Shot Robustness in Vision-Language Models*. arXiv preprint arXiv:2410.21802.

[3] Wu, Z., Lim, S., Davis, L., & Goldstein, T. (2020). *Making an Invisibility Cloak: Real World Adversarial Attacks on Object Detectors*. In European Conference on Computer Vision (ECCV). 2

[4] Nguyen, K., Zhang, W., Lu, K., Wu, Y.-H., Zheng, X., Tan, H. L., & Zhen, L. (2024). *A Survey and Evaluation of Adversarial Attacks in Object Detection*. arXiv preprint arXiv:2408.01934. 2

[5] Xue, M., Sun, S., Wu, Z., He, C., Wang, J., & Liu, W. (2020). *SocialGuard: Adversarial Patch Generation for Protecting Deep Image Detectors*. arXiv preprint arXiv:2011.13560.

[6] Liu, H., Yu, Jie., Li, S., Ma, J., Ji, B.. (2022). *Context-Aware Transferable Adversarial Attacks for Object Detection*. arXiv preprint arXiv:2208.08029.

[7] Zhao, J., Xie, L., Gu, Siqi., Qin, Z., Zhang, Y., Wang, Z., & Hu, Y. (2025). *Universal Attention Guided Adversarial Defense Using Feature Pyramid and Non-Local Mechanisms*. Nature. https://www.nature.com/articles/s41598-025-89267-8 2, 3

[8] Wu, J., Pan, K., Chen, Y., Deng, J., Pang, S., & Xu, W. (2025). *Protego: Detecting adversarial examples for vision transformers via intrinsic capabilities.* arXiv preprint arXiv:2501.07044. 2

[9] Islam, C. M., Chacko, S. J., Nishino, M., & Liu, X. (2025). *Mechanistic Understandings of Representation Vulnerabilities and Engineering Robust Vision Transformers.* arXiv preprint arXiv:2502.04679.

[10] Lin, Y., Zhao, H., Ma, X., Tu, Y., & Wang, M. (2021). *Adversarial Attacks in Modulation Recognition with Convolutional Neural Networks*. IEEE Transactions on Reliability, 70(1), 389–401. https://doi.org/10.1109/TR.2020.3032744 3

[11] Aldahdooh, A., Hamidouche, W., & Deforges, O. (2021). *Reveal of Vision Transformers Robustness against Adversarial Attacks.* arXiv preprint arXiv:2106.03734. 3

[12] Shao, R., Shi, Z., Yi, J., Chen, P.-Y., & Hsieh, C.-J. (2021). *On the Adversarial Robustness of Vision Transformers.* arXiv preprint arXiv:2103.15670. 3

[13] Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Kolesnikov, A., Duerig, T., & Ferrari, V. (2020). The Open Images dataset V4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*.

[14] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Li, F.-F. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, *115*(3), 211–252. https://doi.org/10.1007/s11263-015-0816-y

[15] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., & Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems* (Vol. 32). Curran Associates, Inc. Retrieved from https://papers.nips.cc/paper_files/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html

[16] TorchVision Contributors. (2016). *TorchVision: Computer vision library for PyTorch*. Retrieved from https://github.com/pytorch/vision

[17] Kim, H. (2020). *Torchattacks: A PyTorch repository for adversarial attacks*. Retrieved from https://github.com/Harry24k/adversarial-attacks-pytorch

[18] Wightman, R. (2019). *PyTorch image models*. Retrieved from https://github.com/huggingface/pytorch-image-models

[19] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830. Retrieved from https://jmlr.org/papers/v12/pedregosa11a.html