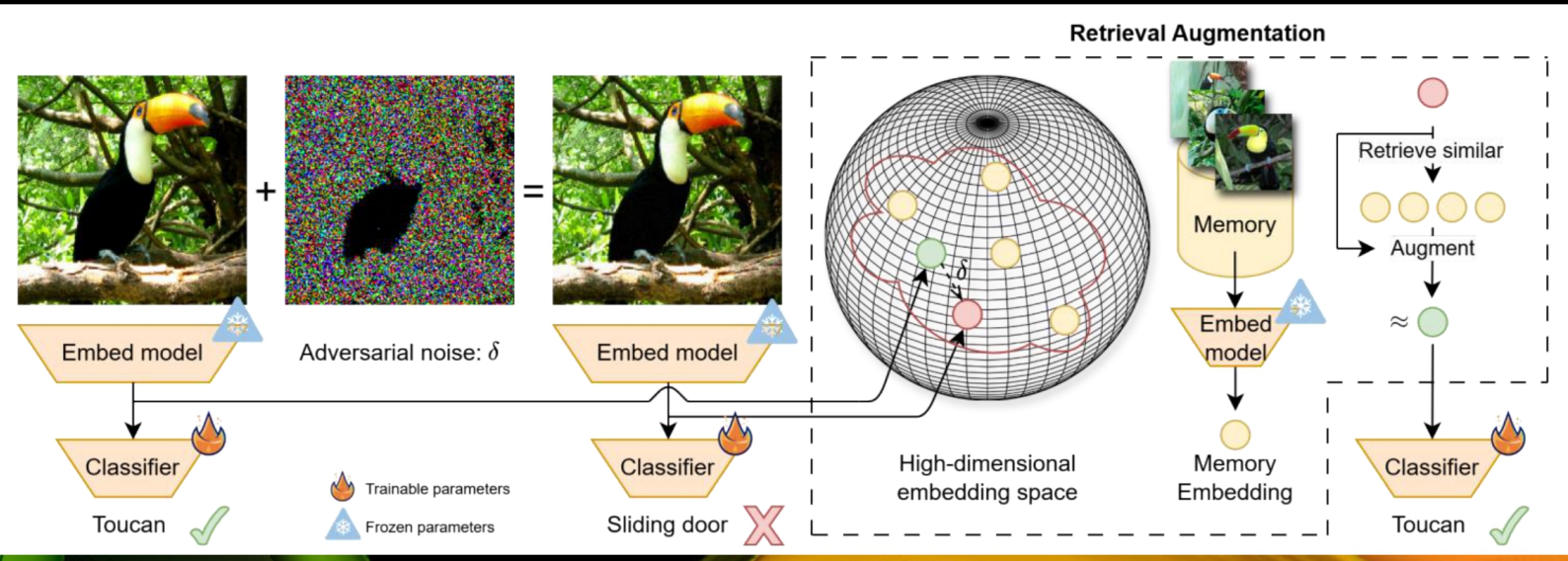
Retrieval-Augmentation for Adversarial Robust Visual Classification

To retrieve or not to retrieve

Olaf Jan Braakman

Delft University of Technology



TU Delft

Retrieval-Augmentation for Adversarial Robust Visual Classification

To retrieve or not to retrieve

by

Olaf Jan Braakman

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday May 9, 2025 at 10:00 AM.

Student number:	4695011	
Project duration:	September 2, 2024 – May 9, 2025	
Thesis committee:	Dr. J.C. van Gemert	TU Delft, Responsible supervisor
	Dr. N.M. Gürel	TU Delft, Daily supervisor
	Dr. S. Dumančić	TU Delft, External commitee member
	S. van Rooij	TNO, External advisor
	Dr. G. Burghouts	TNO, External advisor

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft

Preface

I would like to thank everyone involved throughout my thesis, sharing thoughts, giving feedback and most of all voicing your desire to understand. A special thanks to Sabina and Gertjan from TNO for their guidance throughout the project. Thanks to Merve for being my daily supervisor at the TU Delft and Jan for taking up the role as responsible supervisor on this thesis.

Olaf Jan Braakman
Delft, May 2025

Contents

Nomenclature

Symbols

Symbol	Definition
x	Input vector
y	Class label
f	Classification model
$\mathcal{D}$	Dataset
d	Vector dimensionality
δ	Perturbation
$\|\| \cdot \|\|_p$	L_p norm
ϵ	Perturbation bound
$\mathcal{L}$	Loss function
θ	Model parameters
∇_x	Gradient with respect to x
a	Steps size
x^*	Perturbed input
Π_S	Projection function to set S
f_{RA}	Retrieval-augmented classification model
E	Embedding model
e	Embedding vector
δ_ϵ	Perturbation in embedding space
r	Retrieval augmentation layer
M	Memory set
q	Query vector
k_i	i-th key vector in memory
v_i	i-th value vector in memory (paired with k_i)
w_i	Weight of the i-th key vector in memory
g	Augmentation function
α	Retrieval-augmentation interpolation value
τ	Softmax temperature
$\hat{e}$	Augmented embedding vector

1

Introduction

Our reliance on artificial intelligence for computer vision through deep learning has increased incredibly in the past decade [10]. From self-driving cars [2], facial recognition to unlock your phone [18], or medical imagery [22], deep learning models are able to complete a wide variety of complex image processing tasks. To most of us deep learning technology can feel like a magical black-box that is able to compete with or outperform the best human beings. But as we are starting to rely on AI more and more, can we just blindly trust a decisions an AI makes [19]?

In Figure 1.1 are two seemingly identical no-entry signs taken from a large dataset of images of real world traffic signs [27]. However, a state-of-the-art classification model will make a catastrophic mistake in this case. It misclassifies a no-entry sign for a highway speed limit sign. How is it possible that such a mistake can be made?

Prediction:
No entry sign
(100% confidence)

Prediction
120 km/h sign
(55.25% confidence)

Figure 1.1: Traffic sign misclassification with a seemingly unnoticeable difference between the images.

1.1. Problem Statement

Traditionally computer vision models consisted of human-crafted filters. An example is a basic edge detector algorithm, which finds regions in an image where the intensity 'suddenly' changes [30]. For a hand-crafted algorithm we define the order and types of mathematical operations ourselves. Take as an example the stop sign recognition, it is possible to come up with a set of rules in order to classify the image. E.g. does it contain a red circle of a given size? If so, are there enough white pixels that span a stretched-out rectangle? If both are true, you have detected a no-entry sign. With a rule based algorithm it is possible to deduce how a decision is made, but are these rules general enough to detect a no-entry sign?.

With the adoption of deep learning, computer vision models are only given the input image and a ground-truth output. Engineering these 'filters' is no longer the task for a human, but part of the optimization

task of the deep learning model. Despite no longer relying on human crafted filters or feature detectors, deep learning models like Convolutional Neural Networks (CNNs) [20] with
25 million parameters can surpass human performance on several benchmarks when trained on enough data. Modern computer vision architectures, like the Vision Transformer (ViT) [8] scale to billions of parameters [5]. This trend of scaling model to bilions of parameters has resulted in some of the most capable computer vision models to date.

Deep neural networks consist of an enormous amount of non-linear mathematical operators and possibly billions of tuneable parameters. But this non-linearity has a major drawback: as computer vision models grow larger and become more complex, our human understanding of their internal decision making starts to fade. This lack of explainability of computer vision models, and deep learning models in general, has serious ethical consequences. How can we trust or know that a self-driving car will stop at a red light at a busy intersection? It might do so in a simulated environment or say so for all the test images in a dataset. But what about a change in viewing angle, a subtle shadow, or heavy snow? Is it really possible to consider all of these alternative versions of a red traffic light?

Adversarial machine learning is the study of exploiting model vulnerabilities to alter predictions without being detected. It challenges the trustworthiness of machine learning models and questions their reliability. Interestingly, adversarial examples are ‘poisoned’ images with very small deliberately chosen pixel changes such that a model will wrongly classify it. The traffic sign recognition in Figure 1.1 is an example of such an attacked image. However, this is unacceptable, as a self-driving car is operating is a high-stakes application with serious real-life consequences. It underlines the stark reality of our current understanding of AI models.

1.2. Research Motivation

Fortunately this is not where the story ends. With the advent of Generative Pre-trained Transformers (GPT), large language models have led the charge in model scaling into the billions of parameters. However, training such large models requires enormous amounts of energy and compute time. With the earliest language models it was the case: if your training data changes, you have to train again.

In an ideal scenario a large language model is able to split its factual knowledge from its model parameters. By relying on explicit memory storage a large language model can reduce its number of parameters and we can update its knowledge-base without retraining [21]. Not only that, by retrieving information from an explicit memory, we can observe what information is retrieved in a human understandable format. This makes the retrieval augmentation paradigm an appealing direction in a pursuit for explainability, not only for large language models, but also for computer vision. More over, there have been results for retrieval-augmented large language models indicating that they operate under a lower risk than regular large language models [29, 16].

Motivated by the adversarial vulnerabilities in images and the adoption of the retrieval augmentation paradigm in the large language model and computer vision domain, we arrive at the core of this thesis. In this thesis, we combine these ideas together for a novel defense to adversarial vulnerabilities for computer vision models using retrieval augmentation.

1.3. Research Scope

Based on the robustness properties that retrieval augmentation seem to inhibit, we hypothesize that these robustness properties from the retrieval-augmentation paradigm also transfer to computer vision models. With this hypothesis we arrive at the research question for this thesis:

Does image retrieval augmentation improve robustness against perturbations and adversarial attacks in visual classification tasks?

Additionally we ask the follow sub question: *To what types of perturbations and/or adversarial attacks does image retrieval augmentation improve classification accuracy against?*

This research focuses on the task of visual classification by considering a set of well-established bench-

marking datasets and state-of-the-art image classification models. Adversarial vulnerabilities can be exposed by two types of adversarial attacks which assume different degrees of knowledge about the model itself. For both the white-box and black-box cases we select hand-pick adversarial attacks which are considered standard and/or ‘strong’ as of writing.

1.4. Impact

With this work we underline that existing adversarial vulnerabilities scale to state-of-the-art computer visions and complex image classification tasks. As retrieval-augmentation is an efficient and scalable model adaptation, retrieval augmentation as a defense to adversarial risks has the potential to scale to modern architecture and dataset scales, where so far other defense methods have struggled to.

Adversarial perturbations pose a serious threat to the robustness and trustworthiness of deep learning model. In this thesis we contribute to our shared understanding of deep learning-based computer visions models. It is hard to imagine a world without artificial intelligence anymore. Therefore we hope this work inspires more responsible and ethical use of artificial intelligence in the computer vision domain and beyond.

1.5. Thesis structure

This thesis consists of two parts:

1. **Background**: This is supplementary material for a non-expert reader. It consist of two parts providing background about our two main research motivations: Adversarial AI for computer vision and retrieval augmentation. In the first part, we give a brief explanation of how adversarial attacks work internally, how existing defenses try to mitigate these issues and how they are used in the paper. In the second part we introduce the concept of retrieval augmentation and how it has been applied in large language models and computer vision.
2. **Scientific article**: In this paper, we answer the aforementioned research questions. We explore the retrieval augmentation paradigm as an adversarial defense on a diverse number of datasets to empirically analyze how our defense scales in terms of the number of classes and dataset size. On top of that we compare our method to existing defense mechanisms and perform a hyper-parameter analysis of our retrieval augmentation module. Finally, we experiment with different model and highlight the potential of leveraging information from a different modality.

   Our method and results show that it is possible to adapt an existing deep learning computer vision with a training-free retrieval augmentation pipeline and make it more robust to adversarial input without severe loss of standard classification accuracy.

2

Background

This background chapter consists of two parts:

- Adversarial attacks and defenses (Section 2.1)
- Retrieval augmentation for deep learning (Section 2.2)

These core concepts build up to knowledge required for the paper in Chapter 3. In this paper we propose the idea of retrieval augmentation and analyze how it can play a fundamental role in increasing model robustness against adversarial attacks.

2.1. Adversarial Attacks and Defenses

Consider a point $x \in \mathbb{R}^d$ that has a class y. The goal of an adversarial attack is to slightly change this point x to a new point x^* such that a human eye can barely see the difference. This tiny *perturbation* is malicious in nature in such a way that a classification model misclassifies x^*. This phenomenon of an adversarial examples was first mentioned by Szegedy et al. in 2014 [28] and made explicit by Goodfellow et al. [11]. In Figure 2.1 we demonstrate such an adversarial example. The existence of such examples reveals a significant vulnerability in deep models, particularly in safety-critical applications like autonomous driving, facial recognition, and medical imaging. These attacks not only challenge the robustness of models but also raise broader concerns about trust, reliability, and security in AI systems.

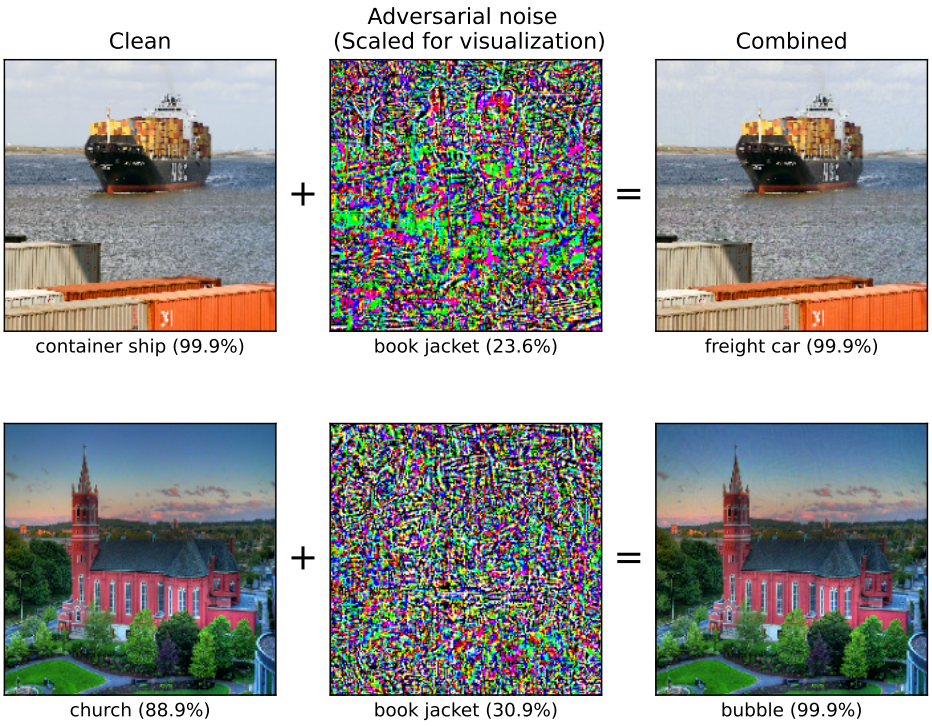


Figure 2.1: PGD adversarial attack visualized on sample images from the ImageNet dataset [7].

To attack a model one needs to know what parts of a model are accessible in order to craft an adversarial example. In literature we commonly differentiate between types: **white-box** attacks and **black-box** attacks.

This section introduces essential mathematical definitions for adversarial attacks, notable attack strategies, and adversarial defenses.

1. **White-box**: White box attacks assume we know everything about a model, from its inputs, model, architecture, model weights, and outputs. In most cases this means we have access to the gradient values in the model.
2. **Black-box** Unlike white-box attacks that require knowledge of the model's internals, black-box attacks assume limited or no access to model parameters. These attacks often rely on transferability. An adversarial examples generated for one model may fool another. A query-based optimization estimates gradients through model queries to craft adversarial inputs. Black-box attacks highlight the real-world feasibility of adversarial threats.

2.1.1. Formal definitions

Before moving to attacks, we need a mathematical framework to define how an attack can operate and under what constraints. As mentioned earlier our data point x has a class label $y \in \mathcal{Y}$ and we consider a classifier $f : \mathbb{R}^d \to \mathbb{R}^{|\mathcal{Y}|}$ which predicts for a data point the corresponding class. We give the definition of an adversarial attack as follows:

Definition 1 *Adversarial Attack*
Under an adversarial perturbation mapping $\mathcal{A} : \mathbb{R}^d \to \mathbb{R}^d$ the original datapoint x is transformed to $x^* = \mathcal{A}(x)$, such that for our classification model:

$$f(\mathcal{A}(x)) \neq f(x)$$

misclassifies x^*

This is a general description of an adversarial mapping. For adversarial perturbations on images we consider the additive adversarial model. Under an additive transformation we ensure the image remains an image and the transformation is mathematically more convenient to work with.

Definition 2 *Additive Adversarial Attack*
Under an additive adversarial attack the original datapoint x is transformed to $x^* = x + \delta$ where $\delta \in \mathbb{R}^d$ is a perturbation, such that for our classification model:

$$f(x + \delta) \neq f(x)$$

misclassifies x^*

To make sure the perturbation remains small we constrain the magnitude of δ to a specific norm. δ is bounded by the L_p-norm, which limits the freedom of δ. Using the L_p norm we can formally define a bound on the perturbation δ. In the context of perturbations for images we often consider the following bounds:

Definition 3 L_2 *Norm*
For $p = 2$, the L_p-norm of x is often refered to as the Euclidean norm is defined as:

$$||x||_2 = \sqrt{|x_1|^2 + |x_2|^2 + \cdots + |x_n|^2}$$

Definition 4 L_∞ *Norm*
For $p \to \infty$, the L_p-norm of x is often refered to as the maximum norm is defined as:

$$||x||_\infty = \max(|x_1|, |x_2|, \ldots, |x_n|)$$

What do these norms mean in terms of input images? When we consider x to be an image, the L_p-norm bounds the change in pixel space to some value often refered to as ϵ. For example, under the L_2 norm the square root of the sum of squares of all pixel changes ϵ cannot be exceeded. For L_∞ it can be interpreted as, the maximum change in pixel value for any pixel in the image can not be higher than ϵ. Formally the bound is defined as:

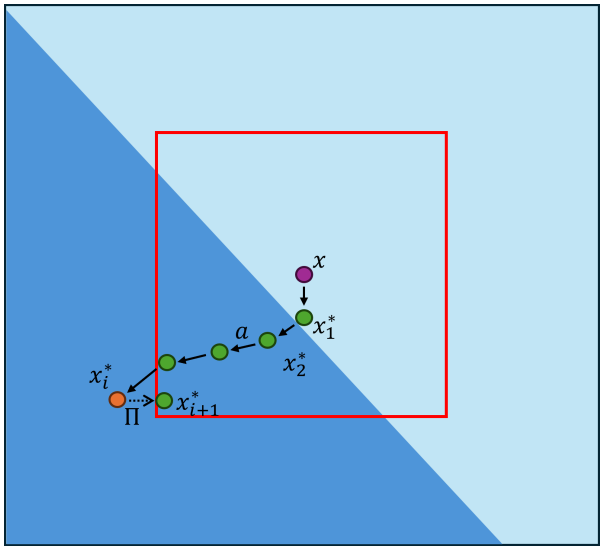


Figure 2.2: Visualization of a decision boundary and how an iterative attack like PGD finds an adversarial example without violating the perturbation bound by projecting it back to the allowed set S, with projection operator Π_S.

Definition 5 *Perturbation bound*
A pertubation $\delta \in \mathbb{R}^d$ is bounded by an L_p norm such that it does exceed a threshold $\epsilon \in \mathbb{R}$

$$||\delta||_p \leq \epsilon$$

For an L_∞-norm bound ϵ for $||\delta||_p \leq \epsilon$, one typically sees it represented as a fraction: $\epsilon = \frac{4}{255}$. A pixel value in in a color channel or for gray-scale images are represented by an 8-bit value which ranges from 0 to 255. In this example it means a maximum of pixel value change of four.

2.1.2. White-Box Attacks

The first white-box attack to leverage this gradient information was the Fast Gradient Sign Method (FGSM) [11]. FGSM generates adversarial examples by performing a one-step gradient update in the opposite direction of the gradient.

$$x^* = x + \epsilon * \text{sign}(\nabla_x \mathcal{L}(\theta, x, y))$$

Given a gradient ∇_x of the loss function of a classification model $\mathcal{L}(\theta, x, y)$, where θ are the accessible model weights, x the input and y the output. By stepping in the 'opposite' direction, or formally the sign, the attack tries to maximize the loss of the model in order to fool the model into misclassifying x^*

Basic Iterative Method

A very simple improvement to FGSM was the Basic Iterative Method (BIM) [9]. The key idea is to apply the FGSM multiple n times with a small step size a and to clip the pixel values to the ϵ bound at each intermediate step. This clipping operation forces that the maximum perturbation δ does not exceed the maximum norm $||\delta||_\infty \leq \epsilon$.

$$x_0^* = x$$
$$x_{i+1}^* = \text{clip}(x_i^* + a * \text{sign}(\nabla_x \mathcal{L}(\theta, x, y)), x - \epsilon, x + \epsilon)$$

Projected Gradient Descent Attack

Projected Gradient Descent (PGD) [24] is a more rigorous extension to BIM. Instead of a clipping operation, the PGD uses a projection function Π_S, which maps the perturbed input back to the allowable

set S, which is constrained by the more general L_p norm. Where BIM initializes x_0^* to the original image, PGD starts at a random initialized location inside of the allowed L_p norm.

$$x_{i+1}^* = \Pi(x_i^* + \alpha * \text{sign}(\nabla_x \mathcal{L}(\theta, x, y)))$$

In Figure 2.2 we visualize a decision boundary and the maximum allowed perturbation in red. For simplicity we assume the PGD attack starts at the original point x. With the small step a the perturbation evolves and in case it violates the ϵ bound it gets projected back inside the allowed set. The resulting adversarial examples for real images are shown in Figure 2.1.

PGD has established itself as a highly effective and go-to attack. It is theoretically well-grounded and at the same time easy to implement. Therefore, PGD is used a benchmark for white-box attack for adversarial examples in computer vision.

Carlini & Wagner Attack

The Carlini & Wagner (C&W) Attack [3] is a powerful white-box attack created in response to a defense called defensive distillation. With this new attack the authors show that these seemingly defended models remain vulnerable to the C&W attack.

It sets itself apart from the iterative attacks like PGD, because the optimization objective for C&W is to find the smallest perturbation δ that causes a misclassification. By minimizing a special objective function z that encourages misclassification under a box constraint for $x + \delta$ given a hyperparameter c.

$$\begin{aligned} &\text{min}_\delta ||\delta||_p + c \cdot z(x + \delta) \\ &\text{subject to } x + \delta \in [0, 1]^d \end{aligned}$$

Optimizing under a box constraint is not a supported operation under gradient descent solvers like Stochastic Gradient Descent (SGD). Through a change of variable the authors make a differentiable approximation of the constraint by optimizing for w by defining:

$$x + \delta = \frac{1}{2}(\tanh(w) + 1)$$

Since the range of the $\tanh$ function is defined for 0 to 1, just like the original constraint $x + \delta \in [0, 1]^d$, it is possible to use an advanced optimizer like Adam.

The C&W attack is defined for L_1, L_2 and L_∞ and shows strong adversarial examples using fewer pixel perturbations to the input image than e.g. PGD. However, generating such examples comes at a higher cost. It should also be noted that there is no explicit upper ϵ bound for this attack.

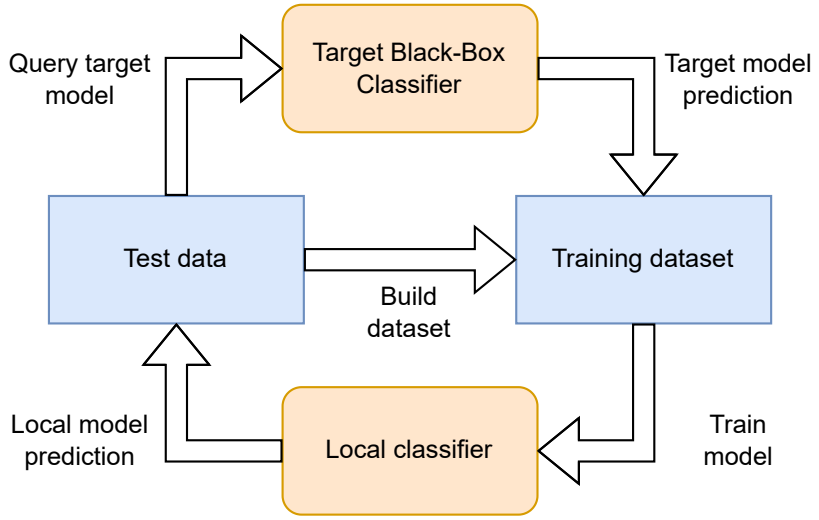


Figure 2.3: Overview of black-box model extraction. Train a local surrogate model that approximates the target model by generating data samples and querying the target model.

2.1.3. Black-Box Attacks

Black-box attacks assume internal information about the model is not directly observable. Black-box attacks simulate a much more realistic scenario for real-world vulnerabilities. A black-box attack can only get new information from the target model by continuously querying it. On top of a perturbation bound, usually a black-box is given a query budget. This limits the number of inference runs on the target model.

Transfer-based attacks

One approach is to try and train a local substitute model and use it to generate adversarial examples that transfer to the target model. In Figure2.3 we show the active learning of a local classifier by continuously querying a black-box target model with test data. By treating the target model as an oracle, i.e. the behaviour we want to mimic, the local model approximates the decision boundaries of the target model.

With a local model it is possible to apply white-box attacks. Surprisingly enough these attacks can transfer to the target model [6] and pose a serious risk. This is even the case if the two models have a different architecture or are trained on different training sets, so long the task is the same [26].

Score-based attacks

Score-based attacks assume to have access to the models input and the predicted confidence scores. These types of attacks skip the surrogate model and try to directly estimate the gradient of a target classifier.

An example of this is the zeroth order optimization (ZOO) [4]. ZOO can efficiently craft adversarial examples by using stochastic coordinate descent. Basically this method tries to approximate the of a single pixel at a time by slightly changing that pixel up and down to observe how the confidence of the model changes. The gradient estimate for a single pixel at index j can be approximated as follows:

$$\frac{\partial f(x)}{\partial x_j} \approx \frac{f(x + ae_j) - f(x - ae_j)}{2a}$$

Here e_j is a zero vector except for a one at pixel index j.

By combining the gradients for all pixels ZOO can perform an estimated gradient descent step with a small step size a. A bunch of optimizations like subsampling the number of pixels, reducing dimensionality of the image, or importance sampling increase the effectiveness and query efficiency of score-based black-box models.

Decision-based attacks

Decision-based attacks assume the least amount of information. Where score-based attack can query to classification probabilities, a decision-based attack only has access to the final decision or prediction.

An efficient black-box decision-based attack example is the Square Attack [1]. It is a trial-and-error approach where squares are randomly placed across the image. With top-1 class label predictions it first tries to misclassify. It follows a simple update rule to find a minimal perturbation that fools the model (Algorithm 1).

Algorithm 1 Square Attack Update Rule (Non-Targeted)

Input: Clean image x, true label y, adversarial example x_i^*, perturbation budget ϵ
Output: Updated adversarial example x_{i+1}^*
Randomly select a square region $R \subseteq$ image domain
Sample a random perturbation δ_R supported on R
Set $x_{\text{candidate}} = \text{clip}(x_i^* + \delta_R, x - \epsilon, x + \epsilon)$
if model$(x_{\text{candidate}}) \neq y$ **then**
    if $\|\delta(x, x_{\text{candidate}})\| < \|\delta(x, x_i^*)\|$ **then**
        $x_{i+1}^* \leftarrow x_{\text{candidate}}$ ▷ Accept the candidate
    else
        $x_{i+1}^* \leftarrow x_i^*$ ▷ Reject, stay at current
    end if
else
    $x_{i+1}^* \leftarrow x_i^*$ ▷ Label not flipped, reject
end if
return x_{i+1}^*

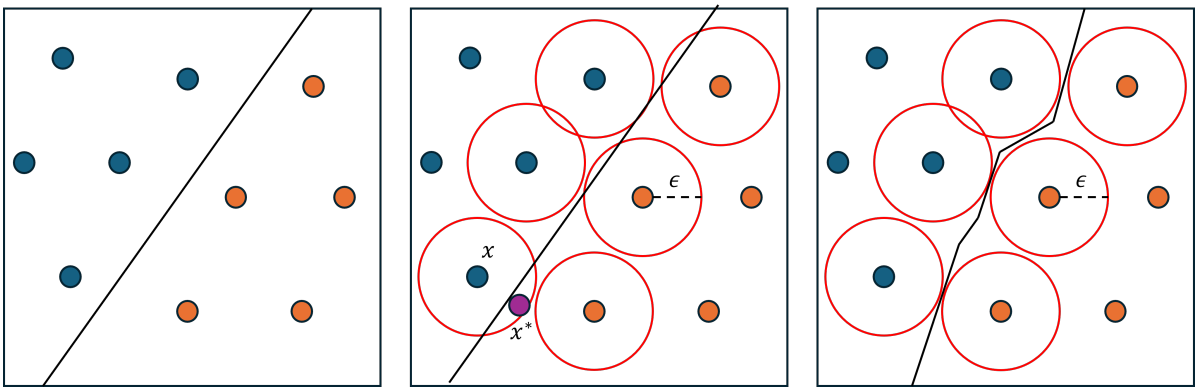


Figure 2.4: Certified smoothing

2.1.4. Adversarial Defenses

In the previous section, we introduced adversarial attacks and how they can fool machine learning models into making incorrect predictions. To address these vulnerabilities, researchers have developed several defense strategies aimed at either increasing the robustness of models or detecting adversarial examples.

Adversarial Training

A common defense is adversarial training. It involves augmenting the training set with adversarial examples during training. The standard training objective tries to minimize the loss of the classifier with respect to its parameters θ (Equation 2.1a). In adversarial training the training objective is formulated as a min-max optimization problem. Maximize the loss under an allowed perturbation $\delta \in \Delta$, but at the same time minimize the loss with respect to the model parameters (Equation 2.1b).

$$\text{Standard Training:} \quad \min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\mathcal{L}(f_\theta(x), y)\right] \tag{2.1a}$$

$$\text{Adversarial Training:} \quad \min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta\in\Delta} \mathcal{L}(f_\theta(x+\delta), y)\right] \tag{2.1b}$$

Practically adversarial training is done as follows. Given a training sample $x, y \sim \mathcal{D}$ from the training set $\mathcal{D}$, craft an adversarial perturbation δ under some bound ϵ that maximizes the loss of the current model. In practice, PGD is often used due to its cost efficiency.

Certified defenses

Unlike adversarial training, which empirically defends against known attacks like PGD, certified defenses aim for formal guarantees that no adversarial example exists within a region around each input. Certified defenses give a guarantee that a model's prediction will remain unchanged under any perturbation within an L_p-norm with bound ϵ.

$$f(x+\delta) = f(x) \text{ for all } ||\delta||_p \leq \epsilon$$

In Figure 2.4 we show a classification task between two classes. An ordinary model can find a decision boundary that perfectly separates the two classes (on the left). However, under a certain certification bound we would like to guarantee that for each point the classification remains the same. But in the middle plot this is certainly not the case. A certified decision boundary separates the classes with the aforementioned guarantee (on the right).

An example of a certified defense is randomized smoothing. The key idea is to train a neural network f with Gaussian data augmentation at variance σ^2. A smoothed classifier f_{smooth} is obtained by returning the most likely class by f with the input x being corrupted by Gaussian noise with variance σ^2. To estimate the most likely prediction for the smoothed classifier, Monte Carlo sampling is used.

Interestingly, the smooth classifier f_{smooth} is provably robust within an L_2 norm around the original input x, implying that for any perturbation δ under the constraint $||\delta||_2 \leq \epsilon$, $f(x + \delta) = f(x)$.

2.1.5. Summary

The exploration of white-box and black-box attacks underscores the critical vulnerabilities inherent in machine learning models. White-box attacks, such as PGD, and the C&W attack, exploit full access to model internals to craft precise and effective adversarial examples. These methods highlight the need for models to be robust against gradient-based manipulations and emphasize the importance of defensive strategies that can withstand iterative and optimization-based attacks. In contrast, black-box attacks, including transfer-based, score-based, and decision-based methods, operate under more realistic constraints where model internals are inaccessible. These attacks demonstrate that even with limited information, adversaries can successfully generate adversarial examples that transfer across different models or exploit decision boundaries through clever querying strategies. The effectiveness of these attacks underscores the necessity for models to be resilient against a variety of query-based and transfer-based threats.

2.2. Retrieval Augmentation

Large Language Models (LLMs) have taken the world by storm. Their wide applicability, ability to condense text, explain topics have changed the way we as humans interact with text information. However, these models are trained on billions of pieces of text and require enormous amounts of power to train and run inference. Pushing for bigger models with more and more parameters, it becomes economically infeasible to retrain a model if part of the training data changes.

To address this problem researchers have tried to decouple the explicit knowledge from the model parameters in an attempt to reduce model size. The power-house in modern large language models facilitating this decoupling is called: **Retrieval Augmentation**.

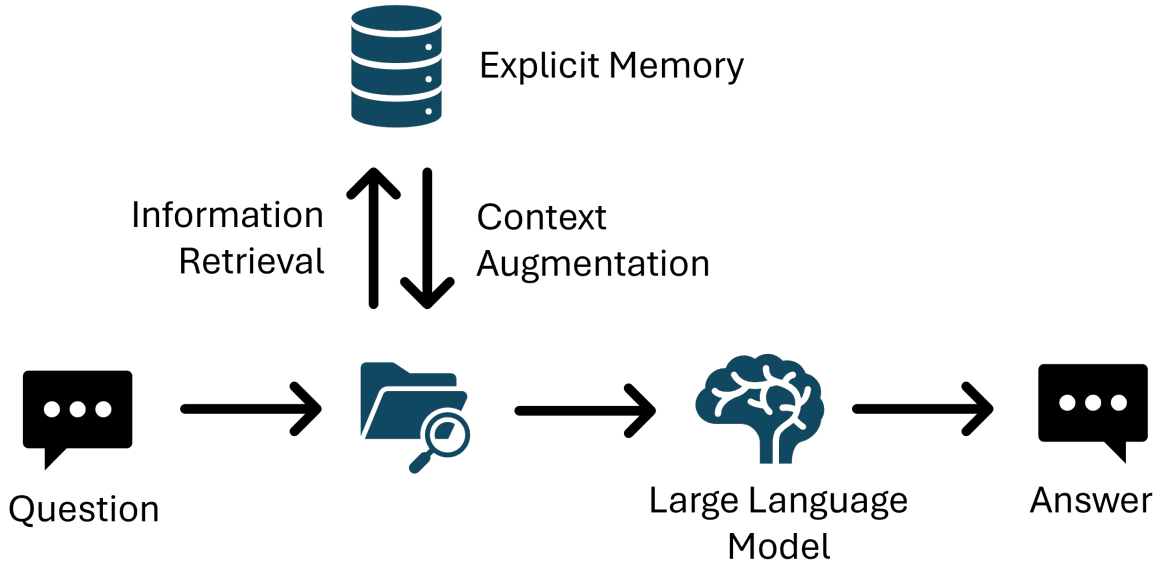


Figure 2.5: Core idea of retrieval augmentation in large language models

2.2.1. Explicit knowledge storage

Traditional neural networks, despite their success in pattern recognition tasks, have historically struggled with algorithmic problems that require explicit memory manipulation. Recurrent neural networks (RNNs) and their variants like Long Short-Term Memory networks (LSTMs) [14] were capable of modeling sequences, but their ”memory” was constrained to fixed-size hidden states. This limitation made it difficult for such models to perform tasks that required dynamic storage and retrieval of arbitrary amounts of information, such as copying, sorting, or associative recall.

Graves et al. introduced the Neural Turing Machine (NTM). It proposes a hybrid model that augments a neural network controller with a differentiable external memory matrix [12]. The controller learns to read from and write to memory via soft attention mechanisms, allowing the system to store explicit knowledge outside of its internal parameters.

This approach shifted the paradigm: instead of relying solely on internal weight-based memory, models could dynamically access an expandable memory during inference. NTMs demonstrated that external memory, when properly integrated, enables neural networks to learn algorithmic tasks and generalize them to novel inputs. This insight established the foundation for subsequent developments in memory-augmented models, retrieval-augmented language models, and broader efforts to separate knowledge storage from reasoning mechanisms in modern machine learning.

2.2.2. Retrieval Augmentation for Large Language Models

Dense Passage Retrieval

How can a model find similar text passages from a memory given a text query q? In Dense Passage Retrieval (DPR) [17], both the query and the passages in a memory set are encoded into fixed-size dense vectors using a bi-encoder architecture, like BERT. The model retrieves passages by calculating the similarity between the query embedding and the passage embeddings in a high-dimensional space. This similarity score is typically computed using the **cosine similarity** between the query and passage embeddings:

Definition 6 *Cosine Similarity*
For two vectors a and b of equal dimension d, the cosine similarity is defined as the normalized dot product between the two vectors.

$$\textit{sim}(a, b) = \frac{a \cdot b}{||a||_2||b||_2} = \frac{\sum_{i=1}^{d} a_i b_i}{\sqrt{\sum_{i=1}^{d} a_i^2}\sqrt{\sum_{i=1}^{d} b_i^2}}$$

The query q is encoded using a query encoder, producing a query embedding E_q. Similarly, each passage p (both relevant and irrelevant) is encoded using the passage encoder to produce the passage embedding E_p. Typically embeddings are already normalized, that is why the normalization step is usually omitted .

In order for the model to learn what are relevant and irrelevant passages, the goal of training is to ensure that the embedding for the relevant passage E_{p^+} is closer to the query embedding E_q than the embeddings for irrelevant passages E_{p^-}.

This is achieved during training. A query q is paired with:

- **Positive Passage** p^+, which is relevant to the query.
- **Negative Passages** p^-, which are not relevant to the query.

The model is trained using a **contrastive loss function** that encourages the relevant passage p^+ to have a higher similarity to the query q than the irrelevant passages p^-. The loss function is defined as

$$\mathcal{L} = -\log \frac{\exp(\textsf{sim}(E_q, E_{p^+}))}{\exp(\textsf{sim}(E_q, E_{p^+})) + \sum_{p^-} \exp(\textsf{sim}(E_q, E_{p^-}))}$$

This binary cross-entropy loss ensures that the relevant passage is ranked higher than the irrelevant ones. DPR has shown significant improvements over traditional sparse retrieval methods like BM25, especially when integrated into systems like open-domain question answering, where retrieval of relevant knowledge from a large corpus is critical. The dense retrieval approach allows for more precise matching of queries to documents by capturing deeper semantic meaning, rather than relying solely on lexical overlap.

Retrieval-Augmented Generation

With a powerful text-to-text retrieval model like DPR as the retrieval engine, Retrieval-Augmented Generation (RAG) [21] enhances language generation models by explicitly incorporating this retrieved information.

In RAG, a retrieval module first fetches a set of relevant documents from a large corpus given an input query. These retrieved documents are then provided to a generative language model, typically a sequence-to-sequence model, which conditions its output not only on the input query but also on the retrieved context.

Formally, given a query q, the model retrieves a set of k documents $\mathcal{P} = \{p_1, \ldots, p_k\}$ from an external memory M and generates an output y by modeling:

$$P(y \mid q) = \sum_{p \in \mathcal{P}} P(y \mid q, p) P(p \mid q)$$

where $P(d \mid q)$ is the retrieval score (e.g., similarity between query and document), and $P(y \mid q, d)$ is the likelihood of generating the output conditioned on both the query and document.

This design decouples factual knowledge from the model's parameters, enabling the knowledge base to be updated independently of the language model. It results in better performance on knowledge-intensive tasks by leveraging up-to-date, retrieved information rather than relying solely on model memorization.

End-to-End Retrieval Augmentation

So far, we have seen how RAG can store knowledge in an external memory. However, the optimization algorithm of a neural network works under the premise that every operation from input x to output y is differentiable. Backpropagation is defined as a method for updating model parameters by computing gradients of the loss with respect to the model's parameters. This requires each operation in the forward pass to be differentiable, so that the gradients can flow backward through the model. The retrieval process itself is typically not differentiable, as it involves discrete operations (e.g., selecting the top-k most relevant passages). This poses a challenge: how can we train a model end-to-end if the retrieval mechanism is not differentiable?

One common approach is to treat the retrieval process as a soft attention mechanism, where the model learns to attend to a continuous weighted combination of passages rather than selecting a discrete number of passages. This allows the retrieval step to become differentiable, enabling the gradients to flow through the entire pipeline during training. In REALM [13, 15], the retriever is trained jointly with the generator to retrieve passages that are most likely to be helpful for answering a given query, and the generation model is conditioned on these passages to generate relevant outputs.

Formally, let q represent a query and D represent the corpus of documents. The retriever in REALM [13] is a function $\text{Retriever}(q; \theta_R)$, where θ_R are the parameters of the retriever. The retriever outputs a set of passages $P_q = \{p_1, p_2, \ldots, p_k\}$ based on the query q. The generator then uses the retrieved passages P_q to generate a response y conditioned on both the query q and the passages:

$$P(y \mid q, P_q) = \text{Generator}(q, P_q; \theta_G)$$

where θ_G are the parameters of the generator. During training, the model is optimized using a joint objective that encourages both the retriever and generator to work together effectively, typically using a combination of contrastive and generation losses.

The retriever in REALM does not select a discrete top-k set of passages. Instead, it generates a continuous distribution over the entire corpus. The passages P_q retrieved for a given query are represented as a weighted sum of document embeddings:

$$P_q = \sum_{i=1}^{|D|} \alpha_i \cdot p_i$$

where α_i represents the soft weight (probability) assigned to each passage p_i, and p_i is the embedding of the i-th passage. The weight α_i is computed as:

$$\alpha_i = \frac{\exp(\text{score}(q, p_i))}{\sum_{j=1}^{|D|} \exp(\text{score}(q, p_j))}$$

where $\text{score}(q, p_i)$ is a similarity function, typically the dot product between the query embedding $\mathbf{q}$ and passage embeddings $\mathbf{p_i}$. This softmax function ensures that the weights sum to 1, and the retriever is differentiable with respect to α_i.

The generator component of REALM then generates a response y conditioned on both the query q and the retrieved passages P_q. The generator is a neural language model, such as BERT or GPT, which is conditioned on the retrieved passages:

$$P(y \mid q, P_q) = \text{Generator}(q, P_q; \theta_G)$$

where θ_G are the parameters of the generator.

2.2.3. Retrieval Augmentation for Computer Vision

In computer vision, retrieval augmentation follows a similar philosophy to that in language models: external memory is used to complement a model's internal representation

Image-to-Image Similarity

Analogous to text retrieval in DPR, image retrieval systems aim to find images similar to a query image in an embedding space. Each image is encoded into a dense vector using a vision backbone (e.g., a convolutional neural network or vision transformer). Given a query image embedding e_q, similarity is computed against a database of image embeddings $\{e_p\}$, often using cosine similarity or Euclidean distance:

$$\text{sim}(e_q, e_p) = \frac{e_q \cdot e_p}{\|e_q\|_2 \|e_p\|_2}$$

The closest images according to this metric are retrieved as candidates for downstream tasks such as classification, captioning, or few-shot learning.

Recent self-supervised learning methods like DINOv2 [25] have demonstrated that vision transformers (ViTs) can learn highly generalizable visual representations without the need for labeled data. DINOv2 pre-trains a ViT model by maximizing the similarity between different augmented views of the same image, while minimizing the similarity between different images.

Because DINOv2 produces semantically meaningful embeddings, it naturally serves as a strong backbone for retrieval tasks. An image embedding generated by DINOv2 can be compared directly against a database of embeddings.

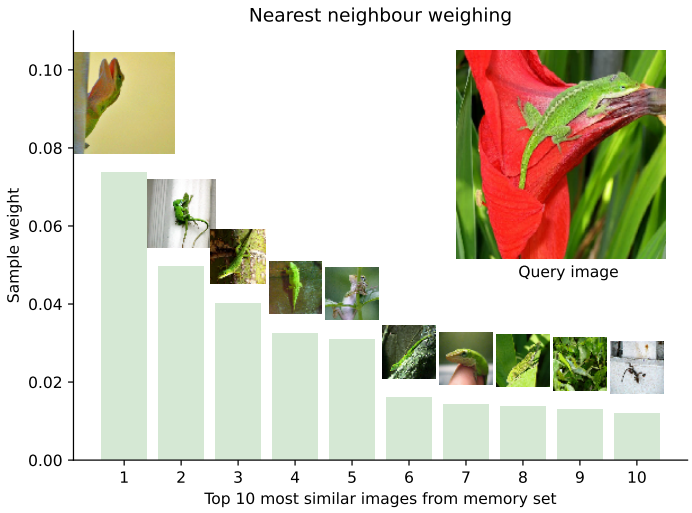
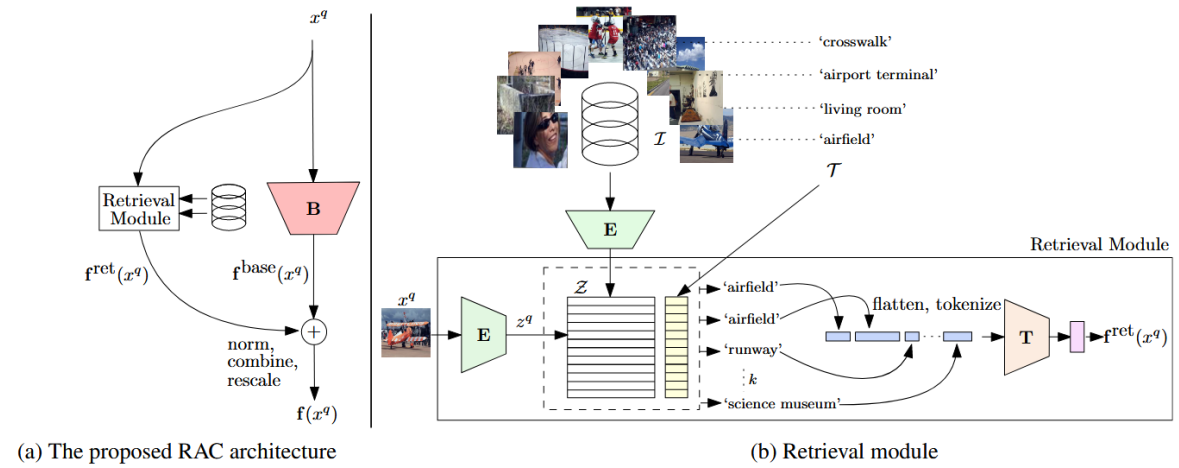


Figure 2.6: Image-to-image retrieval for an image from the ImageNet [7] dataset. Images are weighted based on their cosine similarity to the query image.

Retrieval Augmentation for Image Classification

An example of a direct application of retrieval augmentation in computer vision is retrieval-augmented classification (RAC). Given an input image, the system retrieves k similar images from an external memory bank. These images all have a corresponding image caption which are in turn combined and fed into a BERT-like text encoder. The logit predictions of the original image classification model are combined with the logit predictions of the text encoder (Figure 2.7). This demonstrated the first use of retrieval augmentation for visual classification tasks.



(a) The proposed RAC architecture (b) Retrieval module

Figure 2.7: Image taken from the RAC paper [23]. In (a) the retrieval augmentation extension of the base classification model B. A retrieval module retrieves top-k similar images from a memory set. (b) The associated top-k image labels are combined and fed into a BERT-like text encoder T. Finally the logits are augmented together to make a final classification prediction.

This retrieval-augmented strategy improves performance especially when training data is limited, when classes have few examples, or when dealing with open-world or long-tail classification problems. Retrieval provides additional text context and examples, allowing the model to better infer the correct label under different dataset distributions.

3

Research paper

The scientific article as part of this thesis.

Retrieval Augmentation for Adversarial Robust Visual Classification

Olaf Braakman
Delft University of Technology and TNO Netherlands

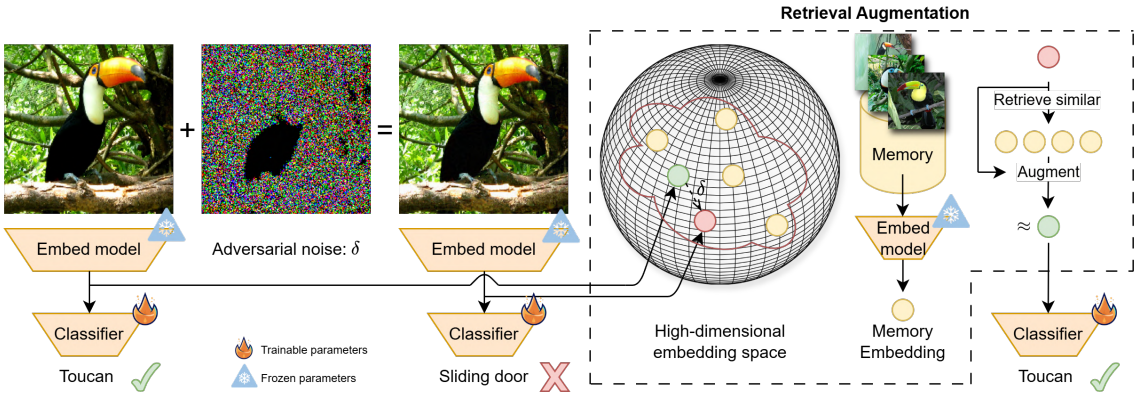


Figure 1. Using retrieval augmentation we show how a pretrained embedding model can be adapted to be more robust against white and black box adversarial attacks.

Abstract

State-of-the-art models are susceptible to adversarial attacks. These attacks can cause catastrophic misclassification when robustness is required. With the increasing popularity of the retrieval augmentation paradigm in deep learning, we adopt it as a fully differential framework for adversarial robustness. We evaluate our method on three visual classification datasets, including ImageNet and attack our model with two white box attacks and a black box attack under various L_2 and L_∞ norms. The results indicate that a robust classifier emerges if the model fully relies on retrieved examples. We find that we can already obtain a PGD robust ImageNet classifier with 80.1% clean and 64.7% adversarial accuracy, using only one or two examples per class from the training data in the memory set. Contrary to other adversarial defense mechanisms, our method works directly on top of pre-trained models and remains robust when other defenses start to degrade for PGD attacks increasing in strength. Code is available at: https://github.com/OlafBraakman/robust-retrieval-augmentation

1. Introduction

Adversarial robustness remains a critical challenge for computer vision models. Even the state-of-the-art models are susceptible to subtle and carefully crafted input perturbations, referred to as *adversarial attacks* [59]. Although adversarial inputs are almost indistinguishable from the original with the human eye, they can cause catastrophic misclassifications [1]. From a more general perspective, robustness in deep learning refers to a model's ability to maintain stable performance under varying conditions, including noise, environmental changes, and adversarial attacks. In literature, this has led to a continuous cat-and-mouse game between attackers and defenders, where attacks continuously find new vulnerabilities and exploits. The goal of adversarial defenses is to reduce the adversarial success rate of these attacks. The most common defense strategy is adversarial training, during which the model is exposed to adversarial examples to help it learn more robust representations. While effective against known attack types, adversarial training is computationally expensive and often reduces performance on clean, unperturbed inputs.

The problem of adversarial robustness is not unique to computer vision models [36]. This is also true for large language models. Interestingly, Retrieval-Augmented Gen-

eration [40], or RAG for short, has surged in popularity due to its simplicity and improvements to text generation quality. Instead of trying to learn all knowledge implicitly in the model weights, RAG offloads this to an indexable external memory. By relying on explicit knowledge storage, RAGs are more consistent compared to standard large language models. With an eye on adversarial robustness, only up until recently has retrieval augmentation been shown to reduce the adversarial success rate in in-context learning for large language models [65]. On top of that Kang et. al prove and show that retrieval augmentation achieves a lower generation risk compared to a standard large language model [33].

RAG brings a lot of benefits to large language models. In turn, this begs the question of to what extent these benefits might transfer to computer vision as well. As pre-trained image encoders become more sophisticated, with vision transformer (ViT) models from the DINO family [11, 47] and CLIP [52], image-to-image search lends itself more to the retrieval augmentation paradigm. Hence retrieval augmentation is actively being explored in the computer vision domain [68]. Retrieval augmentation has been used to increase tail-class classification accuracy and overall classification accuracy with a memory set that is orders of magnitude larger than the training set [30, 31, 42]. Retrieval augmentation in computer vision has been shown to improve classification accuracy and robustness to data scarcity. Retrieval augmentation has been used for a small ResNet-18 model to defend against adversarial attacks under a non-differentiable retriever [66]. Retrieval-augmented vision models have not yet been systematically evaluated under adversarial attack settings for large-scale models and datasets under a full differentiable retrieval pipeline, leaving an important gap that we aim to address.

In this paper, we take the first steps towards answering the question: *Does image retrieval augmentation improve robustness against perturbations and adversarial attacks in visual classification tasks?* And additionally, *what types of perturbations and/or adversarial attacks does image retrieval augmentation improve against?*

We propose an adversarial defense that builds on the retrieval augmentation formula that works directly on top of pre-trained computer vision models. Contrary to existing adversarial defenses, only the classification head needs to be trained. We perform adversarial robustness evaluation on three different visual classification datasets with different backbones. The goal of this research is to take the first steps and demonstrate the potential of retrieval augmentation as a defense against adversarial attacks in the computer vision domain. Our main contributions are as follows:

- We propose a fully differentiable retrieval augmentation layer for pre-trained computer vision models, enhancing robustness against adversarial perturbations without retraining the base model and without drastically compromising accuracy.
- We apply the fully differentiable retrieval-augmented model on three classification tasks and evaluate robustness against white-box and black-box adversarial attacks and compare our work to existing defenses.

2. Background

2.1. Adversarial Attacks

Adversarial attacks refer to small, carefully designed perturbations to input data that cause machine learning models to produce incorrect outputs. These adversarial samples were first documented by Szegedy et al. [59]. These perturbations are often imperceptible to humans but can significantly degrade model performance. They expose critical weaknesses in model generalization and are particularly concerning in high-stakes applications where certification is required. Formally, a model f is considered robust at input x if for any perturbation δ within a bounded norm-ball, the model prediction remains unchanged:

$$f(x+\delta) = f(x), \text{ where } ||\delta||_p \leq \epsilon \tag{1}$$

Here, $\epsilon > 0$ controls the maximal strength of the perturbation and $||\cdot||_p$ denotes the L_p-norm. Where the choice of the L_p-norm determines the perturbation freedom across the image. In literature the L_p-norm can be divided into three categories (Table 1).

Table 1. List of L_p-norm and corresponding interpretation for images in computer vision

L_p-norm	Interpretation
L_1	Under the L_1 bound the attack is constrained by the summed absolute difference of all pixels in the image.
L_2	Contrary to the L_1-norm, the L_2-norm constraints the attack to the the squared difference of all pixels.
L_∞	The L_∞ bound only restricts the maximum pixel change. In literature it also known as the max norm.

Most adversarial attacks assume that they have access to model internals, such as architecture, parameters, and gradients. This allows the attacker to optimize perturbations via backpropagation. These attacks are referred to as: *White-box* attacks. This includes attacks like: FSGM [22], PGD [44], C&W [9]. More details about the attacks used in this

research are in Section 5.3. However, many deployed systems operate under *Black-box* settings, where internal information of the model is inaccessible. Black-box models operate purely on model outputs. Transfer-based attacks [48] exploit the transferability of adversarial examples across models by crafting them on a surrogate model. Score-based attacks, like ZOO [12], estimate gradients via finite differences using only confidence scores. Decision-based attacks, such as Boundary Attack [7], require only the predicted class label and iteratively refine adversarial examples by navigating the decision boundary. More recent methods like Square Attack [2] use randomized search with low query budgets, making them efficient for high-dimensional inputs. Most of all the most important reason to use Black-box models is to reveal gradient obfuscation. This is a phenomenon where defenses rely on masking or distorting gradients rather than improving true robustness. Techniques like input randomization, non-differentiable preprocessing, or shattered gradients can deceive gradient-based methods into failing, giving the illusion of robustness. However, when attacked using gradient-free methods, such defenses often collapse, revealing that the model remains vulnerable. [3, 10, 60].

2.2. Image-to-Image retrieval

Image-to-image retrieval aims to find visually or semantically similar images given an input query image. Early retrieval systems relied on hand-crafted feature descriptors such as SIFT [43], SURF [6], and HOG [16]. These descriptors capture low-level features like keypoints and gradients, but do not generalize well. The advent of deep learning shifted image retrieval towards learned representations. Convolutional neural networks (CNNs) [39] pretrained for classification tasks like ImageNet [17] were shown to produce feature embeddings that were much more suited to image retrieval settings [4]. Self-Supervised Learning (SSL) further expanded retrieval capabilities without requiring labeled datasets. Contrastive loss methods such as SimCLR [13] and MoCo [24] trained models to produce instance-discriminative features, improving robustness to data augmentations and distortions. These SSL models demonstrated that representations learned without labels can perform competitively on retrieval tasks. More recently, DINO [11] and DINOv2 [47] have pushed the frontier of self-supervised image-to-image retrieval. By using vision transformers (ViTs) trained with self-distillation, these models learn dense, semantically meaningful embeddings that are robust to natural image variations. DINOv2 is pretrained on a large set of datasets, including ImageNet-A [29], ImageNet-C [27], ImageNet-R [28]. These datasets are noisy, perturbed, and distribution shifted versions of ImageNet. In this way DINOv2 achieves strong performance on a range of retrieval benchmarks without requiring any labeled fine-tuning, establishing it as a state-of-the-art model for robust self-supervised retrieval.

3. Related work

3.1. Adversarial robustness

Not only can attacks fool a model they can be optimized to target a specific output class as well. Defenses are required to prevent disastrous misclassifications.

- *Adversarial training*: The goal of adversarial training is for a classifier to generalize to adversarial examples as well as clean examples. Adversarial includes an attacker in the training loop to generate adversarial training examples [22, 38]. Madry et al. proposed the multi-iteration projected gradient descent algorithm (PGD) [44], which has become the baseline method of adversarial training [67]. Adversarial training has proven effective against the attack types used during training but has drawbacks. Adversarial training is computationally expensive, it often reduces clean accuracy, and it tends to overfit specific threat models. Overall this leads to poor generalization to unseen attacks [5].
- *Randomized smoothing*: The fundamental idea behind randomized smoothing is to create a smoothed classifier by applying Gaussian noise to a base classifier [14]. This method provides robust theoretical guarantees, making it a popular choice among certified defenses. *Certified defenses* aim to provide formal guarantees under specific threat models, but typically scale poorly to high-dimensional data and complex tasks. Despite recent improvements, randomized smoothing methods remain computationally expensive and worsen existing scalability issues.For a robust prediction at inference time it requires multiple noisy passes through the model which is an additional bottleneck. The curse of dimensionality further limits performance, making scalability the main barrier to broader adoption [37].
- *Other strategies*: Other strategies include input preprocessing [51], ensemble methods [26, 58, 60], feature denoising [41, 64], and k-nearest neighbors [50, 55, 61]. Ensemble methods enhance robustness by aggregating predictions from multiple diverse models or checkpoints, reducing the likelihood that all models will be simultaneously fooled by the same perturbation. Feature denoising techniques, on the other hand, aim to suppress adversarial noise in the intermediate representations of the network.

3.2. Retrieval augmentation in computer vision

Recent literature demonstrates that computer vision models also benefit from retrieval augmentation in a variety of ways. RAC (Retrieval-Augmented Classification) [42] uses retrieval augmentation by constructing a parallel retrieval module. The memory set contains image and text descrip-

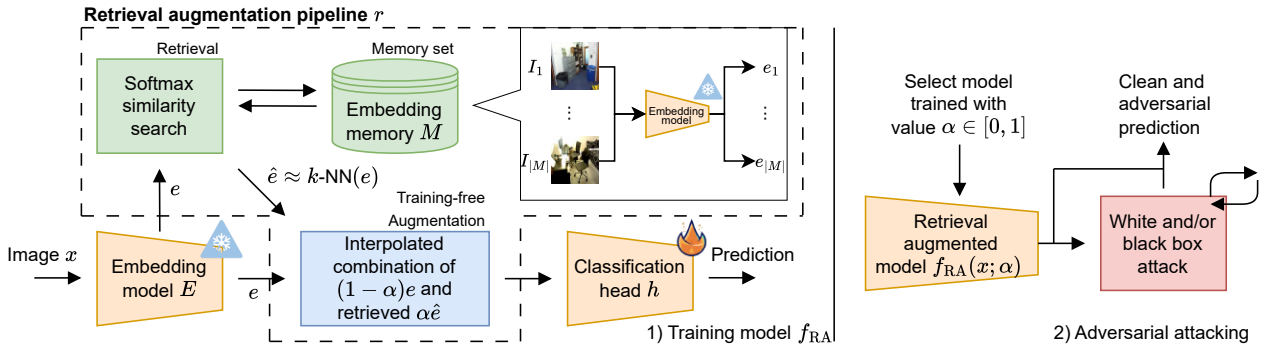


Figure 2. Top-level workflow of the 1) retrieval-augmented model where the memory set is constructed from $|M|$ embedded images. A weighted temperature softmax using the cosine similarity is used as a differentiable nearest neighbor approximation. The summed weighed retrieved embeddings $\hat{e}$ are combined with the input embedding e and fed through a trainable classification head. 2) At evaluation time we analyze for different values of α how the adversarial accuracy changes.

tion pairs which can be queried using image-to-image embedding similarity. In the end, k pairs are retrieved and the associated text labels are embedded using a BERT [18] text encoder. The input image is fed through the base network to a vision transformer and resulting logits are combined with the logit predictions of the text encoder. Using this approach the authors demonstrate that RAC models can learn a high accuracy on tail classes. Iscen et al. address the limitations of RAC by expanding the memory set to over a billion image-text pairs and perform the augmentation step at the embedding level [31]. They extend their work by exploring the application of retrieval augmentation for zero-shot CLIP-based [52] vision-language models [30]. Furthermore, retrieval augmentation has been used for other downstream tasks like image captioning [53].

4. Method

We consider a basic visual classification model f consisting of a frozen pre-trained feature extraction model $E : \mathbb{R}^{3\times W\times H} \rightarrow \mathbb{R}^d$, projecting an RGB image of width W and height H to a embedding vector of size d, and a trainable classification head $h : \mathbb{R}^d \rightarrow \mathbb{R}^{|\mathcal{C}|}$ predicting the class logits. We create a retrieved-augmented model f_{RA}, which is an extension of the original model f, but with access to an external memory M of unperturbed image embeddings. We do this by adding a training-free retrieval augmentation pipeline $r : \mathbb{R}^d \rightarrow \mathbb{R}^d$ after the embedding model and before the classification head.

Suppose an input image x is adversarially attacked $x^* = x + \delta$. Then both the image and the perturbation $x + \delta$ undergo a highly non-linear transformation when passed through the embedding model $E(x+\delta) = e+\delta_e$. The magnitude of δ_e depends in part on the strength of the adversarial attack and the adversarial robustness of the embedding model. We assume that if δ_e is small enough, then qualitative retrieval of similar image embeddings is still meaningful. Based on that assumption, we use the perturbed image embedding $e + \delta_e$ and a combination of the retrieved unperturbed image embeddings $\hat{e}$ to construct a more robust embedding representation. The novelty of our method lies in the training-free augmentation step, where we are able to combine the input embedding and retrieved embedding with a blending parameters α and pass it through a classification layer to obtain a more adversarial robust prediction.

After training f_{RA} with a choice of α, we want to empirically investigate that the adversarial robustness condition (Equation 1) holds more often under various adversarial attacks for f_{RA} than for f under different adversarial attacks and perturbation bounds (Equation 2) given a dataset $\mathcal{D}$.

$$\sum_{x_i \in \mathcal{D}} \mathbb{I}\{f_{\text{RA}}(x_i^*) = f_{\text{RA}}(x_i)\} > \sum_{x_i \in \mathcal{D}} \mathbb{I}\{f(x_i^*) = f(x_i)\}$$
$$\text{subject to} \quad \|\delta\|_p \leq \epsilon \tag{2}$$

In layman terms, given a test dataset $\mathcal{D}$, we want to know whether a retrieval-augmented model f_{RA} can be more robust to some maximum allowed perturbation δ than a model f without retrieval augmentation for an adversarial input $x^* = x + \delta$.

We formalize this approach in a retrieval-augmented framework, grounded in three core components: (1) a memory set, (2) a retriever, and (3) an augmentation function. We visually present our adoption of the retrieval augmentation paradigm in Figure 2 and go into detail for each of these components in the next sections.

4.1. Memory set

We define the memory M as a set of image embeddings sampled from the training set of $M \subseteq \mathcal{D}_{\text{train}}$. Ideally, M

contains representative and clean samples. On top, we assume that every class is equally represented in the memory set. For each image x_i we pre-compute and L_2 normalize its embedding $e_i = ||E(x_i)||_2$ for computational efficiency. We refer to an embedding e_i in the memory set as key k_i. Practically, we implement the memory set $M = \{k_1, k_2, \ldots, e_m\}$ as an $|M| \times d$ matrix with the embeddings as column vectors and store it in VRAM.

4.2. Retrieval

The retrieval module is responsible for finding similar embeddings from the memory module. Given a query embedding $||E(x)||_2 = e$, we compute the cosine similarity s_i between each embedding $k_i \in M$ in the memory set as the normalized dot product between two embeddings as follows:

$$s_i = \text{sim}(q, k_i) = \frac{e \cdot k_i}{\|e\|_2 \|k_i\|_2} \tag{3}$$

The cosine similarity captures the change in directionality between two vectors. For identical vectors, the cosine similarity is 1, -1 for complete opposite vectors, and 0 for exactly orthogonal vectors. The use of another distance metric is also possible, however, for high dimensions distance metrics like the Euclidean distance become less meaningful under the 'curse of dimensionality' [49].

Commonly, retrieval systems perform a hard top-k selection to fix the number of relevant documens [31, 40, 66]. However, this operation is non-differentiable. White-box attacks make use of gradient flow through a model. To simulate a full retrieval augmentation pipeline attack we approximate the nearest neighbor algorithm with a differentiable softmax function. To simulate a top-k selection we add a temperature scaling factor $\tau > 0$. By tuning the temperature parameter we can control the sharpness of the weight that is assigned to each memory key. For small τ only the most similar keys are accounted for as the others are suppressed towards zero. We compute the weight w_i for each key k_i in the memory set as:

$$w_i = \frac{\exp(s_i/\tau)}{\sum_j^{|M|} \exp(s_j/\tau)} \tag{4}$$

Finally we compute the weighted embedding mean $\hat{e}$ of all retrieved keys as follows:

$$\hat{e} = \sum_{i=1}^{|M|} w_i k_i \tag{5}$$

In Figure 3 we visualize an example of weighted embedding retrieval with their associated images for an query image on the ImageNet dataset [17]. For visualization purposes we only show the top-10 most similar images.

Previous applications of retrieval augmentation for visual classification have paired images with an additional text modality [30, 31, 52]. In Equation 5 the weighted keys are returned, however, formally one can substitute k_i with another associated value embedding v_i. In the Supplementary work E we substitute the value embeddings with depth image embeddings.

4.3. Augmentation

We introduce an interpolation hyperparameter $\alpha \in [0, 1]$ which allows the model to 'blend' between the original image embedding e and the weighted mean from the memory set $\hat{e}$. In this step we rely on the geometric regularities of the embedding representations [15, 32] to combine embedding vectors together. This modeling decision allows us to gain insight into how robust and how accurate different linear combinations of the original input e and memory set approximation $\hat{e}$ are without introducing any trainable parameters.

$$g(e, \hat{e}) = (1 - a)e + \alpha\hat{e} \tag{6}$$

Note that the original model f is identical to f_{RA} when $\alpha = 0$. For $\alpha = 1$ we turn the problem into a smooth nearest neighbor classification problem.

4.4. Threat model

For our retrieval augmentation defense, we assume the adversary's goal is to cause a non-targeted misclassification of the model. We assume the case where the attack has white-box access to all parameters of the model. The adversary is only able to make small pixel-level adjustments to the input image within L_2 or L_∞ distortion norm. Depending on the dataset we allow up to a maximum perturbation $\epsilon = \frac{16}{255}$. For an overview of attack and dataset hyperparameters see Appendix A. Targeted attacks are out of the scope of this research.

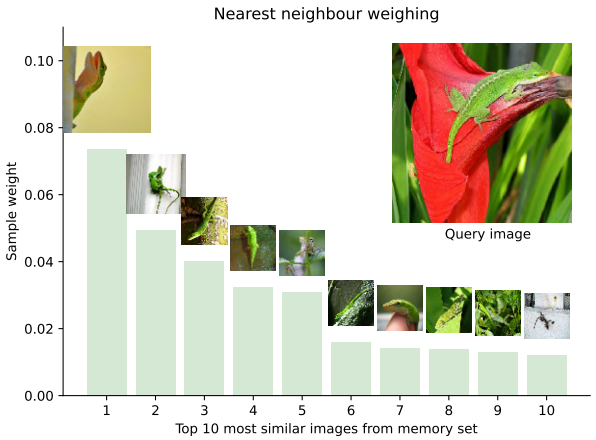


Figure 3. Example of how similar items are retrieved and weighed

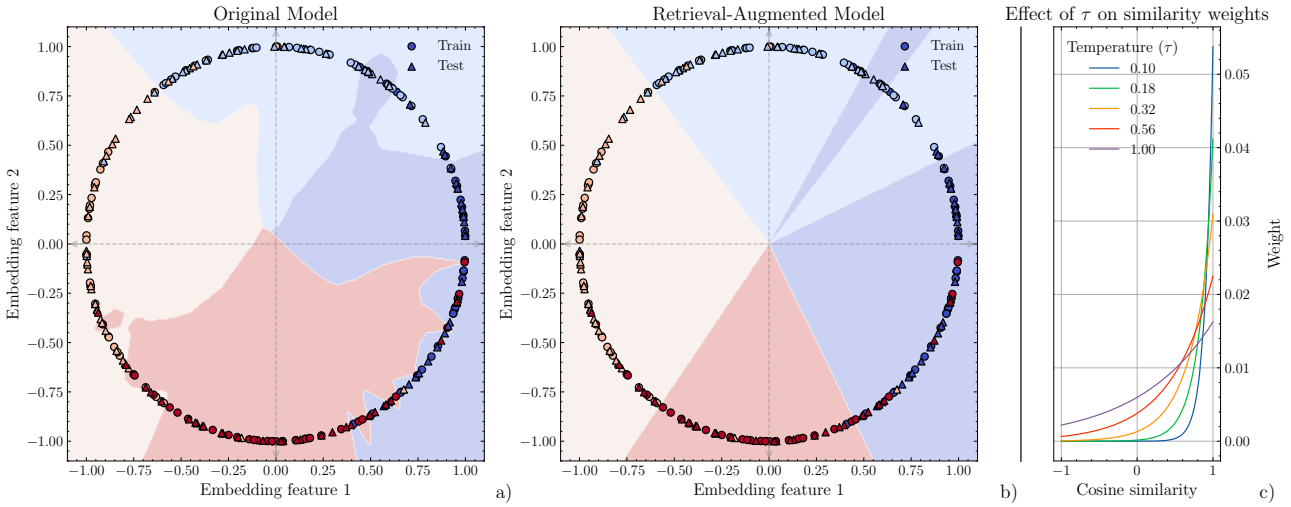


Figure 4. Two-dimensional embedding space example with no a) and full b) retrieval augmentation. c) The effect of different temperatures τ shows the change in the weights of embeddings in the memory set.

Since we use (a subset of) the training data as the memory set we assume it is accessible, but we assume the adversary cannot modify any of the memory set's contents.

4.5. 2D Toy example

The size of the embedding space is too high to visualize. To understand the robustness consequences of an added retrieval augmentation pipeline, we visually analyze a low dimensional example. In this example the embedding space is two-dimensional. Equally for image embeddings, we assume that directionality in this embedding space encodes semantic information. In Figure 4 we generate four classes each in one in the positive and negative x and y axis direction with 0.6 standard deviation. We sample 128 training points and project all embeddings a unit distance from the origin. We train two models: 4a) A model without retrieval augmentation and 4b) A model with retrieval augmentation $\alpha = 1$.

We visualize the decision boundary for both models with the training data used as the memory set. A direct consequence is that the retrieval-augmented model relying fully on the retrieved embeddings has smoothed decision boundaries. On top, the decision boundaries meet in the origin utilizing the directionality as prior information. In the original model the decision boundary does not utilize the prior. Instead, the model learns to overfit towards noisy/mislabeled samples from a different class. This introduces extra decision boundaries along our circle increasing the likelihood for a sample under a small adversarial perturbation to be misclassified. For $\alpha = 1$ it turns the classification head into a soft nearest neighbor classifier. In Figure 14 we show the evolution of the decision boundary given linear values of α. Finally, we show in Figure 4c) how the softmax approximates a hard weighted k-NN for a memory set of size 128. For smaller values of τ more cosine similar items receive a higher weight.

5. Experiments

The flexibility of the method allows it to be applied to different types of pre-trained visual backbones and different datasets.

5.1. Embedding models

To demonstrate how our method works across different model architectures we select three models with unique architecture techniques and training approaches. We first highlight our backbone, dataset and attack choices and then elaborate the experimental setup.

- DINOv2 [47]: DINO, short for self-DIstillation with NO labels, is a self-supervised feature extractor based on the ViT architecture. Compared to supervised vanilla ViTs and ConvNets, DINOv2 obtains more clear explicit semantic information. On top of that, the features are also excellent for instance level classification [63]. We use the distilled ViT-L/14 model pre-trained on ImageNet with 300M parameters which outputs 1024-dimensional features.
- ResNet-50 [25]: To test our work on different architectures we select a convolution network from the ResNet family. We select the ResNet-50 model pre-trained on ImageNet and strip the last layer to obtain a 2048-dimensional feature vector.

Importantly, we freeze the backbone parameters to isolate the effects of our retrieval model.

Table 2. Overview of the datasets used with the number of classes at different orders of magnitude.

Dataset	#Training	#Validation	#Classes
ImageNet [17]	1281167	50000	1000
CIFAR100 [35]	50000	10000	100
GTSRB [57]	31367	7842	43
SUN RGBD [56]	4845	4659	19

5.2. Datasets and memory

We evaluate our method on three benchmarking datasets with high diversity and with class numbers at different orders of scale to see how our method scales (Figure 7). We use the training and validation split of each dataset to train our models and use the training split for the memory set from which the model can retrieve samples. However, due to the computational complexity of the similarity computation running at $O(n^3)$ we set at maximum memory set size at 50,000. We make sure to subsample each class equally.

- *ImageNet (ILSVRC 2017)* [17]: ImageNet is a highly diverse and large-scale dataset consisting of 1,281,167 training images and 50,000 validation images across 1,000 unique object classes. It has been a key driver in advancing the field of visual recognition by providing a robust benchmark for object classification tasks.
- *CIFAR-100* [35]: CIFAR-100 is a lightweight image classification dataset consisting of 50,000 training images and 10,000 test images across 100 fine-grained classes, all at a low resolution of 32×32 pixels. Due to its compact size and complexity, it is widely used for benchmarking lightweight models and evaluating robustness to adversarial attacks.
- *GTSRB (German Traffic Sign Recognition Benchmark)* [57]: The GTSRB dataset contains nearly 40,000 images across 43 unique traffic sign classes. Captured under real-world driving conditions, the images exhibit variations in lighting, occlusion, and motion blur, making it a valuable benchmark for robust traffic sign classification.

5.3. Attacks

We evaluate our model versus a series of adversarial attacks under different L_p-norms. Evaluating robustness under both white-box (gradient-based) and black-box (gradient-free) settings is essential. A robust model must resist diverse attack strategies across different norms, without relying on broken or obscured gradients.

- *PGD (Project Gradient Descent)* [44]: The PGD attack is an iterative adversarial white-box attack method that perturbs inputs within a constrained norm ball, maximizing the model loss. It builds on the Fast Gradient Sign Method (FGSM) [22] by applying multiple small changes which are then projected back into the allowed perturbation range. It is widely regarded as a strong first-order adversary and serves as a standard benchmark for evaluating model robustness.
- *C&W (Carlini and Wagner)* [9]: The C&W attack is a powerful white-box, optimization-based adversarial attack that iteratively searches for adversarial examples capable of bypassing many defenses that are effective against other attacks. Unlike norm-constrained attacks such as FGSM or PGD, the C&W attack does not explicitly enforce a maximum perturbation bound ϵ, but instead minimizes distortion as part of its objective. This typically results in high-quality, low-distortion adversarial examples, though at the cost of slower computation.
- *Square attack* [2]: This is a black-box adversarial attack that operates without any gradient information. By acting only on a model decision, it randomly and adaptively samples and modifies image regions with square-shaped perturbations. Among black-box attacks, it has a high success rate using relatively few queries.

We show examples of each attack in Figures 10, 11, 12 and 13 in Appendix B

5.4. Experimental setup

The setup consists of two RTX 3080Ti graphics cards that we use for classification head training and adversarial attack generation. We freeze the backbone parameters to isolate the effects of the retrieval augmentation layer. The hyperparameters for both model training are listed in Table **??** Appendix A. We use the open-source *torchattacks* [34] implementations of these adversarial attacks. We report the clean classification accuracy across the whole test. Generating adversarial examples across the whole test set for each dataset is computationally too demanding. We approximate the adversarial accuracy in Equation 2 by randomly sampling a subset $\mathcal{D}' \subseteq \mathcal{D}$ of size n. Depending on the computational load of generating an adversarial example we change the size of the subset. Unless reported otherwise, for the white-box attacks we sample $n = 1000$ points and for the black-box attacks $n = 250$. For iterative white-box attacks we use 50 iterations to ensure convergence and for black-box attacks set a limit a 5000 queries.

6. Results

For different values of the interpolation parameter $\alpha \in [0, 1]$, we evaluate the clean and adversarial classification of the retrieval-augmented model $f_{\text{RA}}(\cdot; \alpha)$. Recall that for $\alpha = 0$, the retrieved embeddings are not used, therefore f_{RA} and f are identical. The main results for the ImageNet and CIFAR-100 datasets under the PGD L_∞, C&W L_2, and Square L_2/L_∞ attacks are listed in Table 3. We find that the clean accuracy for ImageNet across the whole test set is highest for $\alpha = 0$ at 82.1% and drops linear to 79.0% for $\alpha = 1$, similarly for CIFAR-100 the clean accuracy drops

Table 3. Comparison of the clean and adversarial accuracy of the retrieval augmentation module for different values of α under different types of adversarial attacks.

Model DINOv2 (ViT-B)	ImageNet					CIFAR-100				
	Clean Acc (%)	Adversarial accuracy (%)				Clean Acc (%)	Adversarial accuracy (%)			
		PGD	C&W	Square attack			PGD	C&W	Square attack	
		L_∞	L_2	L_2	L_∞		L_∞	L_2	L_2	L_∞
$\alpha = 0$ (no retrieval)	**82.1**	0.0	64.0	40.0	3.6	**84.2**	0.0	6.4	63.6	0.4
$\alpha = 0.25$	81.8	0.0	59.4	44.4	5.2	84.1	0.0	5.4	66.6	0.4
$\alpha = 0.5$	80.7	0.0	52.7	45.2	7.2	83.5	0.0	5.4	69.2	0.8
$\alpha = 0.75$	80.4	0.0	48.6	49.2	10.4	81.4	0.1	5.4	67.2	2.0
$\alpha = 0.95$	78.9	0.0	68.0	53.2	28.4	80.3	1.6	22.0	70.0	2.8
$\alpha = 0.99$	78.7	0.0	68.2	54.0	28.0	80.0	2.3	33.8	70.8	2.8
$\alpha = 1$ (full retrieval)	79.0	**64.7**	**77.4**	**72.4**	**59.2**	80.1	**74.7**	**79.9**	**80.4**	**38.0**

from 84.2% to 80.1%. However, in return, we observe that for $\alpha = 1$ the adversarial accuracy remains stable across both the white and black box attacks. Interestingly enough, for increasing α, we do not see a linear increase in adversarial accuracy. For the PGD attack, adversarial classification accuracy is almost zero for all values of α, except $\alpha = 1$. This suggests that adversarial gradients steps can penetrate to the adversarial input embedding despite it being scaled down. Only when fully discarding the adversarial embedding does the model regain classification accuracy. We confirm this by plotting the cosine similarity between the embeddings of clean images and their adversarial variant for the same value of α in Figure 5. We expect that for an increasing value of α the cosine similarity between the two embeddings to decrease, but the results do not suggest such a trend. Only for $\alpha = 1$ do we see a strong similarity between the clean and adversarial embeddings.

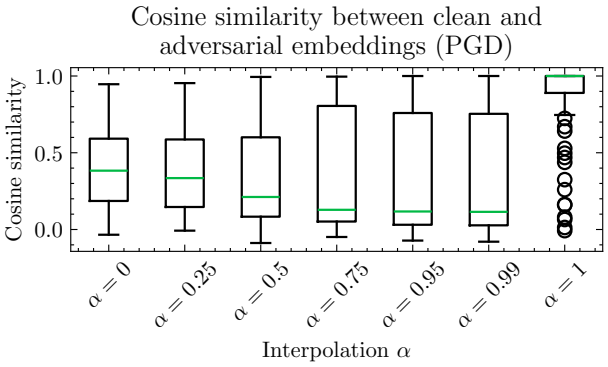


Figure 5. Cosine similarity between the clean embedding e_{clean} and adversarial embedding e_{adv} obtained by $e(1-\alpha)+\alpha\hat{e}$ for different values of α under PGD attack with $\epsilon = \frac{16}{255}$ for ImageNet.

The adversarial accuracy for C&W attack on ImageNet follows a noteworthy pattern. From $\alpha = 0$ to $\alpha = 0.75$, the adversarial accuracy decreases instead of increases. Only with α approaching 1, does the adversarial accuracy increase again. The C&W attack does not constrain the perturbation to a fixed ϵ bound, but instead formulates the attack as an optimization problem that balances minimizing the perturbation size and achieving misclassification. We show the boxplot of L_2 norm for the successful adversarial example generated by C&W for the same values of α in Figure 6. We observe a similar pattern at $\alpha = 0.75$ where the top whisker (third quartile + 1.5 IQR) is lowest and IQR around the median is tight. This suggests that compared to other values of α, the $\alpha = 0.75$ is a unique case where the model is more susceptible to smaller perturbations. This is a finding that contradicts our expectation of a linear trend in adversarial accuracy.

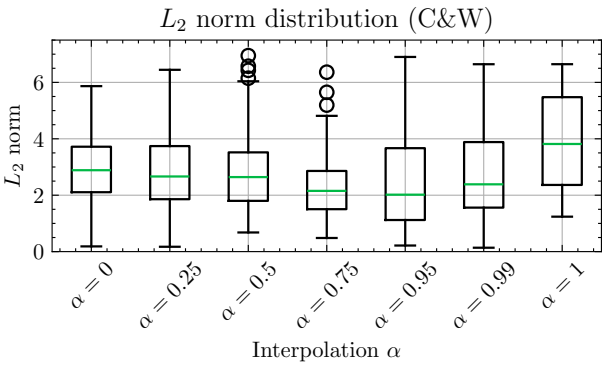


Figure 6. L_2 norm of the perturbation of the adversarial image found by C&W attack for different values of α for ImageNet.

The black-box square attack does show an monotone increase in adversarial accuracy for ImageNet. Under the L_2 bound the retrieval-augmented model outperforms the baseline model with 20 to 30 percentage points in adversarial classification accuracy. However, for CIFAR-100 the L_∞ attack is very successful even under larger values of α. This might be due to the fact that under the L_∞ bound for 32x32

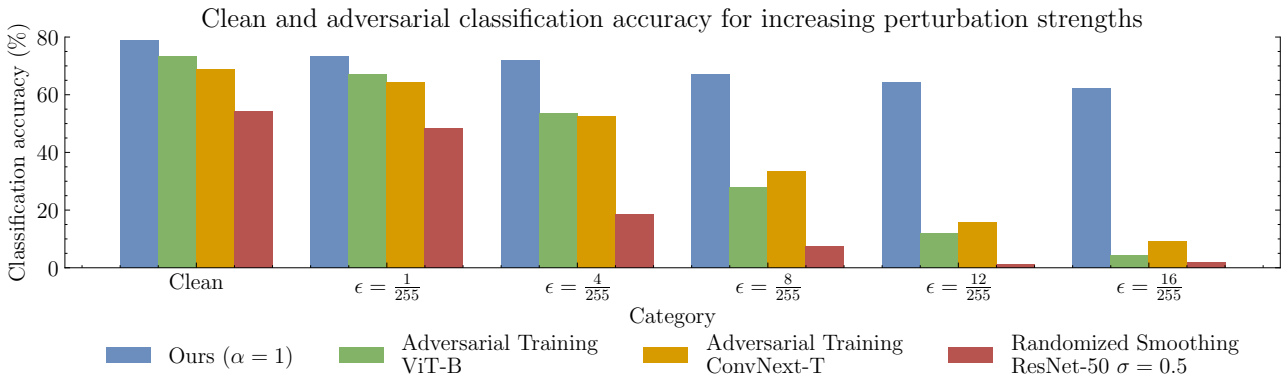


Figure 7. Comparison of the full retrieval model ($\alpha = 1$) with three baseline models under a PGD attack: Adversarial Trained ViT-B, Adversarial Trained ConvNext-T and Randomize Smoothing ResNet-50 architecture. The step size a for the PGD attack is set to $a = \max(\frac{1}{255}, \frac{\epsilon}{4})$.

images the cost of random trial-and-error search is more effective. This result highlights that for small images with a low resolution the retrieval mechanism, even at $\alpha = 1$, struggles to remain robust under L_∞ square attack.

6.1. Baseline comparison

We compare our best model with $\alpha = 1$ against the two most commonly used defense strategies on ImageNet, namely adversarial training and randomized smoothing. For the adversarially trained models we select two different backbone types: The ViT-B architecture adversarially trained up to $\epsilon = \frac{4}{255}$ for 50 epochs and a ConvNext-T model adversarially trained up to $\epsilon = \frac{8}{255}$ [54]. The second baseline is a pre-trained randomized smoothing model with the ResNet-50 architecture with a Gaussian noise standard deviation $\sigma = 0.5$ [14]. We select $\sigma = 0.5$ because it is a balance between clean accuracy and adversarial accuracy. To run the PGD attack on the randomized smoothing model we take the mean of the logits for the number of randomized samples. For computational feasibility, we take $n = 32$ samples per image. In Figure 7 we plot the clean and adversarial accuracy for all the models for increased perturbation strength ranging from $\frac{1}{255}$ to $\frac{16}{255}$. Our retrieval-augmented model is able to withstand much higher perturbations compared to other defenses without having seen any adversarial examples or changing the input images. The decision boundary smoothing with $\alpha = 1$ remains robust against stronger PGD attacks while the baselines degrade.

6.2. Effects of temperature

The temperature parameter τ defines the sharpness of the softmax function which approximates the hard k-NN operation. We show for $\alpha = 1$ on CIFAR-100 that with a decreasing value of τ, the model approaches the original clean classification accuracy. Interestingly for larger values of τ

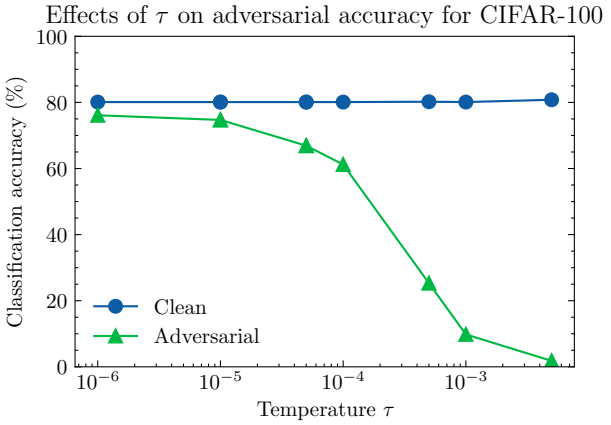


Figure 8. Softmax temperature τ versus clean and adversarial classification accuracy of the PGD attack on CIFAR-100 for $\alpha = 1$ with a memory set of 50,000 samples

the adversarial robustness seems to disappear and the adversarial accuracy reaches zero. For a temperature $\tau \to 0^+$ the softmax function becomes infinite in sharpness.

We hypothesize that a small τ forces the model to weigh only the closest memory sample, which approximates a top-1 nearest neighbor classification which in turn prevents a perturbation δ_ϵ in embedding space to change the embedding weights. However, it should be noted that a

6.3. Memory set size

For all previous experiments, the training set was used as the memory, although capped at a maximum of 50,000 samples. We vary the number of embeddings in the memory set by randomly sampling a subset of the training data, ensuring each class is equally represented. We plot the clean and adversarial classification accuracy for increasing sizes of the

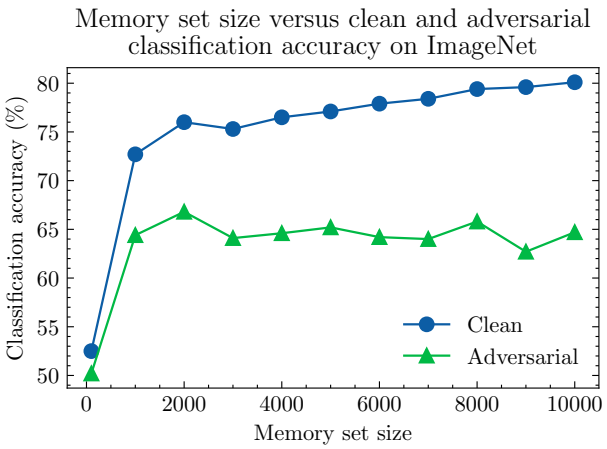


Figure 9. Memory set size variations for CIFAR-100.

memory set for the ImageNet dataset. Notably, the adversarial accuracy does not increase after using more than 2000 training samples in the memory set. This is less than 0.5% of the training set, or roughly one to two samples per class for ImageNet.

Increasing the memory does improve the clean classification accuracy. However, it comes at the cost of increased runtime of $O(n^3)$ for a larger growing memory set. The scaling problem has been well addressed in the literature, with popular techniques including hierarchical softmax [46], noise-contrastive estimation (NCE) [23] and negative sampling [45]. The retrieval can also be applied to a hard k-NN selection and implemented with a library like FAISS [19], but again this would break the gradient flow through the model.

While increasing the memory set size improves clean classification accuracy, it significantly increases the runtime due to the $O(n^3)$ complexity, making it impractical for very large memory sets. Future work could explore the aforementioned approximations or more efficient retrieval methods to address the runtime complexity while maintaining the benefits of retrieval augmentation.

6.4. Choice of backbone

We focus on the GTSRB dataset and train two different backbone architectures: The DINOv2 model (used throughout the rest of the paper) and a ResNet-50 architecture. For the two backbones, we report the clean and adversarial classification accuracy in Table 4 for the same attacks as in Table 3. The results show that even for the ResNet-50 architecture retrieval augmentation provides adversarial robustness. However, the PGD robustness for the DINOv2 model is almost twice that of the ResNet-50, arguably because the DINOv2 is a more robust self-supervised feature extractor trained on a large plethora of data and natural adversarial examples. This is an interesting finding, which suggests that with more robust feature extractors a higher adversarial accuracy can be obtained to global high frequency noise.

Table 4. Comparison between the DINOv2 backbone and the ResNet-50 backbone clean and adversarial classification accuracy for white and black box attacks for the GTSRB dataset.

GTSRB					
	Clean	Adversarial accuracy (%)			
		PGD	C&W	Square att.	
Model	Acc (%)	L_∞	L_2	L_2	L_∞
$\alpha = 0$					
DINOv2	89.4	0.0	0.0	1.6	0.0
ResNet-50	79.3	0.0	1.6	1.2	0.4
$\alpha = 1$					
DINOv2	**90.7**	**61.1**	**83.3**	56.8	**28.0**
ResNet-50	79.5	31.5	71.1	**59.6**	27.2

7. Conclusion and discussion

We propose and demonstrate the first adaptation of retrieval augmentation for robust visual classification across a broad set of datasets, model backbones and pipeline hyperparameters. Compared to other prominent defense mechanisms, retrieval augmentation as a defense remains robust even under higher perturbation bounds for the PGD attack. Under full retrieval conditions, the adversarial accuracy is highest across all datasets. For other hybrid combinations of the original embedding and the retrieved embeddings, the adversarial accuracy does not necessarily follow a linear pattern for different models and different datasets. In these hybrid cases, the retrieval model is still susceptible to adversarial gradients inside PGD, bringing the adversarial accuracy towards zero. Under the C&W attack, adversarial accuracy actually drops when reliance on the retriever increases, before going up again when almost fully relying on the retriever. On the other hand we do observe around a steady 20 to 30 percentage point increase in adversarial accuracy for the black box square attack case.

From an attack perspective, retrieval augmentation as a defense might introduce new vulnerabilities, especially in the retrieval module. We encourage future work to try and attack these specific parts of the pipeline under different threat models. So far we have only looked at uniformly sampled memory sets from the training data, however, it is unclear how different distributions of memory items affect the clean and adversarial classification accuracy. We showed that retrieval augmentation can be done with different modalities, however, the application and benefits for adversarial robustness with multi-modal data is still unanswered.

It must be stated that the literature on existing adversarial

attacks is rich and it is possible to extend evaluation with a plethora of hyperparameters combinations. However, most importantly, the takeaway of this research is the formation of a solid foundation for future work in effective adversarial defenses with differentiable retrieval augmentation. To conclude it is still an open question if formal robustness guarantees for retrieval-augmented model exist under different type of attacks and perturbation strengths.

References

[1] Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. 6: 14410–14430. Conference Name: IEEE Access. 1

[2] Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. 3, 7

[3] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. 3

[4] Artem Babenko, Anton Slesarev, Alexandr Chigorin, and Victor Lempitsky. Neural codes for image retrieval. 3

[5] Tao Bai, Jinqi Luo, Jun Zhao, Bihan Wen, and Qian Wang. Recent advances in adversarial training for adversarial robustness. 3

[6] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded up robust features. In *Computer Vision – ECCV 2006*, pages 404–417. Springer Berlin Heidelberg. Series Title: Lecture Notes in Computer Science. 3

[7] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. 3

[8] Ali Caglayan, Nevrez Imamoglu, Ahmet Burak Can, and Ryosuke Nakamura. When CNNs meet random RNNs: Towards multi-level analysis for RGB-d object and scene recognition. version: 2. 18, 1

[9] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. 2, 7

[10] Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. 3

[11] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. 2, 3

[12] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. ZOO: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 15–26, . 3

[13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, . 3

[14] Jeremy M. Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified adversarial robustness via randomized smoothing. 3, 9

[15] Guillaume Couairon, Matthijs Douze, Matthieu Cord, and Holger Schwenk. Embedding arithmetic of multimodal queries for image retrieval. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 4946–4954. IEEE. 5

[16] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, pages 886–893 vol. 1. ISSN: 1063-6919. 3

[17] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. ISSN: 1063-6919. 3, 5, 7

[18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. 4

[19] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The faiss library. 10

[20] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. ImageBind: One embedding space to bind them all, . 18, 1

[21] Rohit Girdhar, Mannat Singh, Nikhila Ravi, Laurens van der Maaten, Armand Joulin, and Ishan Misra. Omnivore: A single model for many visual modalities, . version: 2. 18, 1

[22] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. 2, 3, 7

[23] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304. JMLR Workshop and Conference Proceedings. ISSN: 1938-7228. 10

[24] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning, . version: 3. 3

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, . 6

[26] Warren He, James Wei, Xinyun Chen, Nicholas Carlini, and Dawn Song. Adversarial example defenses: Ensembles of weak defenses are not strong. . 3

[27] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. 3

[28] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization, . 3

[29] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples, . 3

[30] Ahmet Iscen, Mathilde Caron, Alireza Fathi, and Cordelia Schmid. Retrieval-enhanced contrastive vision-text models, . 2, 4, 5

[31] Ahmet Iscen, Alireza Fathi, and Cordelia Schmid. Improving image recognition by retrieving from web-scale image-text data, . 2, 4, 5

[32] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. 5

[33] Mintong Kang, Nezihe Merve Gürel, Ning Yu, Dawn Song, and Bo Li. C-RAG: Certified generation risks for retrieval-augmented language models. 2

[34] Hoki Kim. Torchattacks: A pytorch repository for adversarial attacks. 7

[35] Alex Krizhevsky. Learning multiple layers of features from tiny images. 7

[36] Pranjal Kumar. Adversarial attacks and defenses for large language models (LLMs): methods, frameworks & challenges. 13(3):26. 1

[37] Anupriya Kumari, Devansh Bhardwaj, and Sukrit Jindal. Rethinking randomized smoothing from the perspective of scalability. 3

[38] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. 3

[39] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. 86(11):2278–2324. Publisher: Ieee. 3

[40] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. 2, 5

[41] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. Defense against adversarial attacks using high-level representation guided denoiser. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1778–1787. IEEE. 3

[42] Alexander Long, Wei Yin, Thalaiyasingam Ajanthan, Vu Nguyen, Pulak Purkait, Ravi Garg, Alan Blair, Chunhua Shen, and Anton Van Den Hengel. Retrieval augmented classification for long-tail visual recognition. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6949–6959. IEEE. 2, 3

[43] David G. Lowe. Distinctive image features from scale-invariant keypoints. 60(2):91–110. 3

[44] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. 2, 3, 7

[45] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. 10

[46] Frederic Morin and Yoshua Bengio. Hierarchical probabilistic neural network language model. 10

[47] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. 2, 3, 6

[48] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. 3

[49] Dehua Peng, Zhipeng Gui, and Huayi Wu. Interpreting the curse of dimensionality from distance concentration and manifold effect. 5

[50] Neehar Peri, Neal Gupta, W. Ronny Huang, Liam Fowl, Chen Zhu, Soheil Feizi, Tom Goldstein, and John P. Dickerson. Deep k-NN defense against clean-label data poisoning attacks. 3

[51] Han Qiu, Yi Zeng, Qinkai Zheng, Shangwei Guo, Tianwei Zhang, and Hewu Li. An efficient preprocessing-based approach to mitigate advanced adversarial attacks. 73(3):645–655. 3

[52] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. abs/2103.00020. 2, 4, 5

[53] Rita Ramos, Desmond Elliott, and Bruno Martins. Retrieval-augmented image captioning. 4

[54] Naman D. Singh, Francesco Croce, and Matthias Hein. Revisiting adversarial training for ImageNet: Architectures, training and generalization across threat models. 9

[55] Chawin Sitawarin and David Wagner. On the robustness of deep k-nearest neighbors. In *2019 IEEE Security and Privacy Workshops (SPW)*, pages 1–7. 3

[56] Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. SUN RGB-d: A RGB-d scene understanding benchmark suite. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 567–576. IEEE. 7, 1

[57] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The german traffic sign recognition benchmark: A multi-class classification competition. In *The 2011 International Joint Conference on Neural Networks*, pages 1453–1460. IEEE. 7

[58] Thilo Strauss, Markus Hanselmann, Andrej Junginger, and Holger Ulmer. Ensemble methods as a defense to adversarial perturbations against deep neural networks. 3

[59] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. 1, 2

[60] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. 3

[61] Ren Wang, Tianqi Chen, Philip Yao, Sijia Liu, Indika Rajapakse, and Alfred Hero. ASK: Adversarial soft k-nearest neighbor attack and defense. 3

[62] Tong Wu, Liang Tong, and Yevgeniy Vorobeychik. Defending against physically realizable attacks on image classification, . 1

[63] Zhirong Wu, Yuanjun Xiong, Stella Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance-level discrimination, . 6

[64] Cihang Xie and Yuxin Wu. Feature denoising for improving adversarial robustness. 3

[65] Simon Chi Lok Yu, Jie He, Pasquale Minervini, and Jeff Z. Pan. Evaluating the adversarial robustness of retrieval-based in-context learning for large language models. version: 1. 2

[66] Jake Junbo Zhao and Kyunghyun Cho. Retrieval-augmented convolutional neural networks against adversarial examples. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11555–11563. IEEE. 2, 5

[67] Weimin Zhao, Sanaa Alwidian, and Qusay H. Mahmoud. Adversarial training methods for deep learning: A systematic review. 15(8):283. Number: 8 Publisher: Multidisciplinary Digital Publishing Institute. 3

[68] Xu Zheng, Ziqiao Weng, Yuanhuiyi Lyu, Lutao Jiang, Haiwei Xue, Bin Ren, Danda Paudel, Nicu Sebe, Luc Van Gool, and Xuming Hu. Retrieval augmented generation and understanding in vision: A survey and new outlook. 2

A. Hyperparameters

Whenever the hyperparameters are not mentioned in the experiments we select the hyperparameters for each attack and dataset as listed in Table 5. Similarly for model hyperparameters in Table 6.

Table 5. White-box and black-box attack hyperparameters used for each dataset.

Pertubation	L_p-norm	Parameters				Black/ White box
		CIFAR-100	ImageNet	GTSRB	SUN RGBD	
PGD	L_∞	$\epsilon = \frac{8}{255}$ $\alpha = \frac{2}{255}$	$\epsilon = \frac{16}{255}$ $\alpha = \frac{4}{255}$	$\epsilon = \frac{16}{255}$ $\alpha = \frac{4}{255}$	$\epsilon = \frac{16}{255}$ $\alpha = \frac{4}{255}$	White
C&W	L_2	$c = 1$	$c = 1$	$c = 1$	n/a	White
Square attack	L_2	$\epsilon = 0.5$ queries $= 5000$	$\epsilon = 3.0$ queries $= 5000$	$\epsilon = 3.0$ queries $= 5000$	n/a	Black
	L_∞	$\epsilon = \frac{8}{255}$ queries $= 5000$	$\epsilon = \frac{16}{255}$ queries $= 5000$	$\epsilon = \frac{16}{255}$ queries $= 5000$	n/a	Black

Table 6. Optimizer parameters used to train the classification head for each dataset.

Hyperparameter	ImageNet	CIFAR100	GTSRB		SUN RGBD
Optimizer	AdamW	AdamW	AdamW	AdamW	AdamW
Learning rate	0.0001	0.0001	0.001	0.01	0.0002
Betas (β_1, β_2)	0.9, 0.999	0.9, 0.999	0.9, 0.999	0.9, 0.999	0.9, 0.999
Weight decay (λ)	0.01	0.02	0.01	0.01	0.03
Epsilon (ϵ)	1e-8	1e-8	1e-8	1e-8	1e-8
Epochs	6	30	15	25	40
Scheduler	Cosine Warmup	Cosine Warmup	Cosine Warmup	Cosine Warmup	Cosine Warmup
Warmup	1	5	2	2	5
Backbone	DINOv2	DINOv2	ResNet-50	DINOv2	SUNRGBD
Embedding dim	1024	1024	2048	1024	1024
Hidden dims	1024	512	1024	512	512
Dropout	0.5	0.5	0.5	0.5	0.5

B. Adversarial examples

We list five adversarial examples for each attack: PGD (Figure 10), C&W (Figure 11), Square attack L_2 (Figure 12) and Square attack L_∞ (Figure 13). We use the base DINOv2 model without retrieval-augmentation to generate these adversarial examples.

B.1. PGD examples

We show the results of the PGD attack for 10 iterations with $\epsilon = \frac{16}{255}$ and step size $a = \frac{4}{255}$. All clean images are correctly classified and the adversarial images are misclassified.

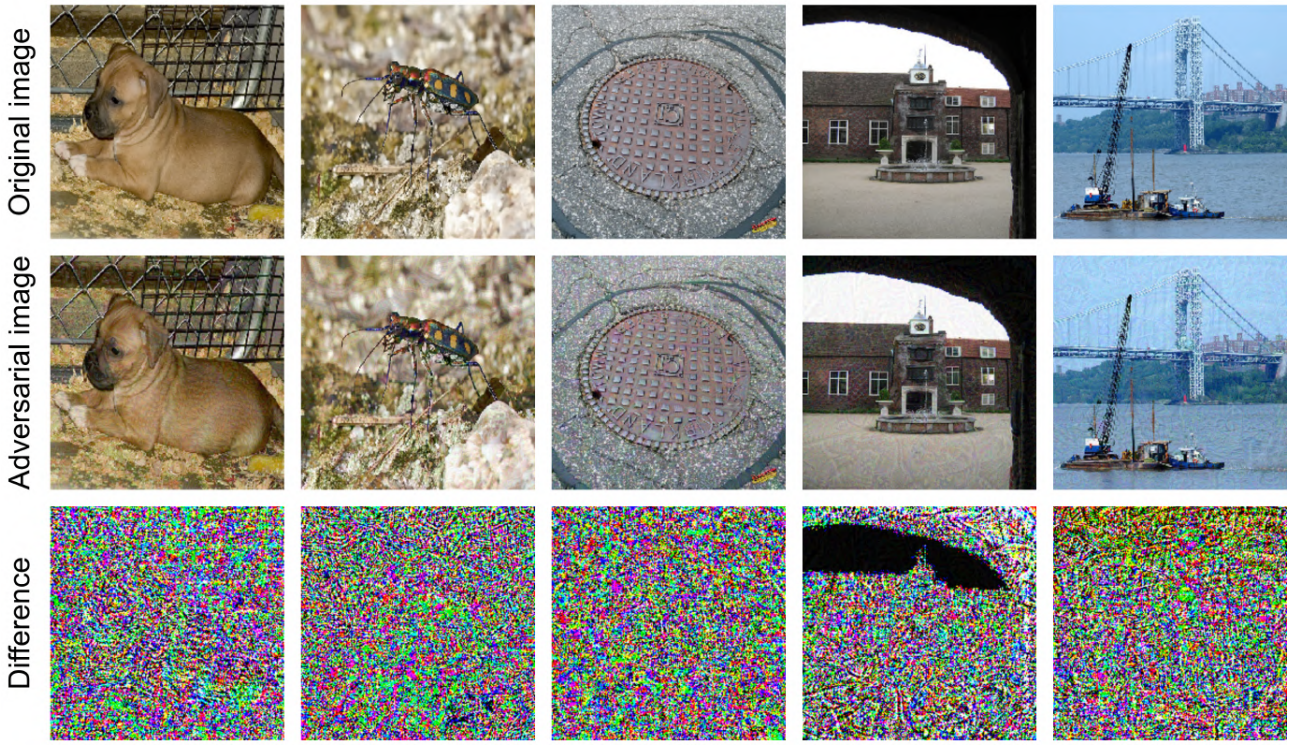


Figure 10

B.2. C&W examples

We show the results of the Carlini and Wagner attack for 50 iterations with $c = 1$. All clean images are correctly classified and the adversarial images are misclassified.

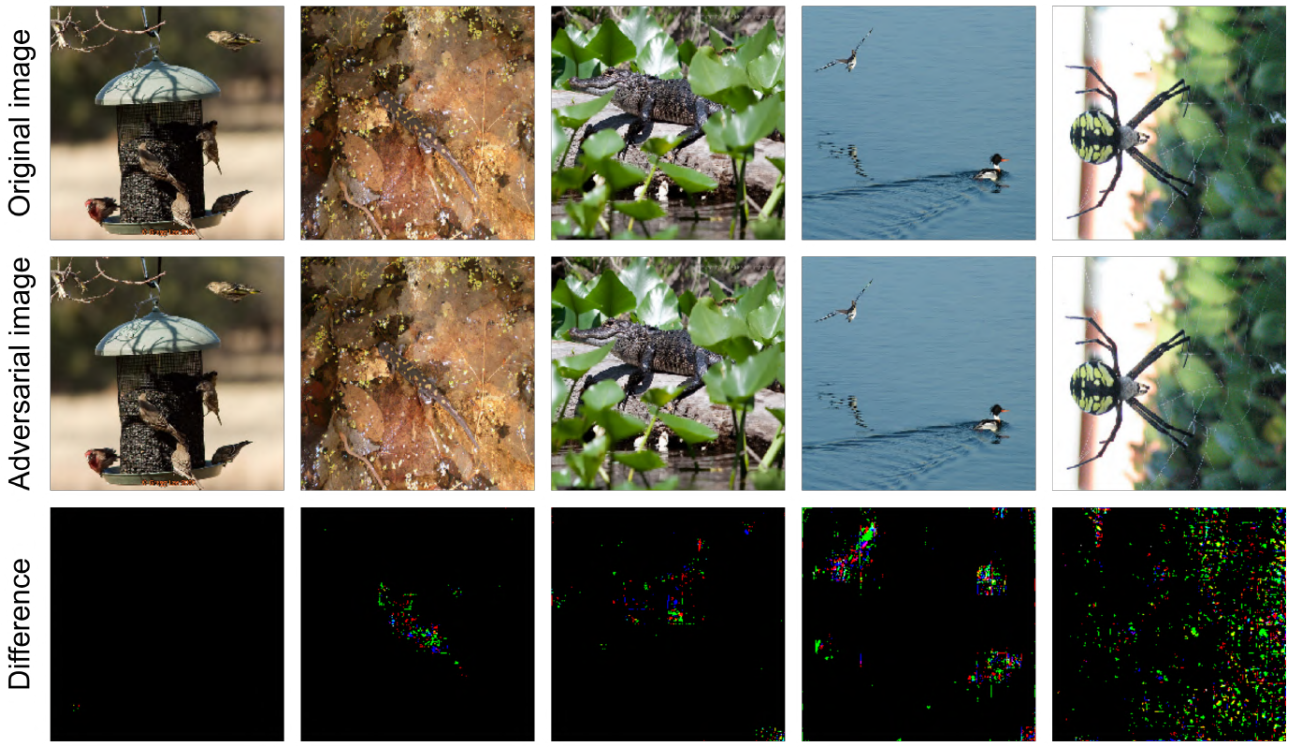


Figure 11

B.3. Square attack L_2 examples

We show the results of the Square attack L_2 attack with $\epsilon = 3.0$ for the first three and $\epsilon = 12.0$ for the fourth and fifth image. The black box attack has a budget of 5000 queries. All clean images are correctly classified and the adversarial images are misclassified.

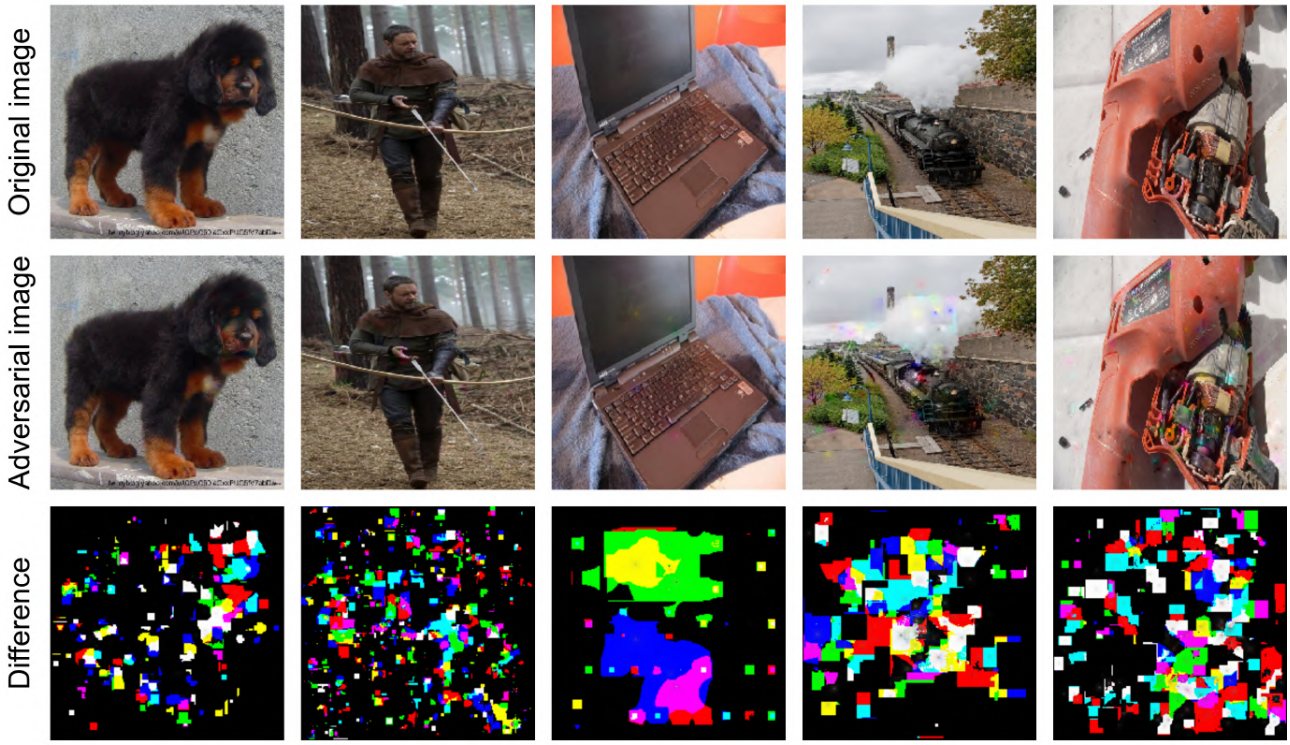


Figure 12

B.4. Square attack L_∞ examples

We show the results of the Square attack L_∞ attack with $\epsilon = \frac{16}{255}$ and a budget of 5000 queries. All clean images are correctly classified and the adversarial images are misclassified.

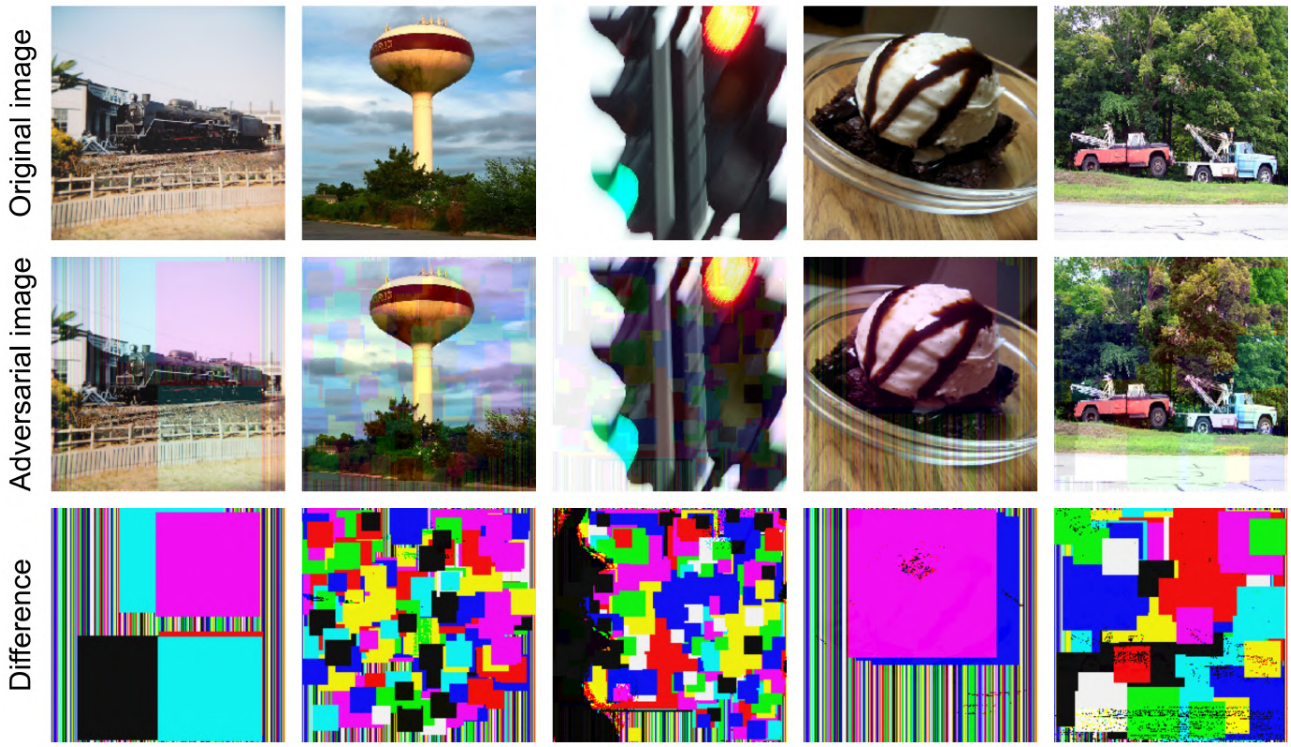


Figure 13

C. Data preparation

To use the SUN RGBD dataset for a classification task we use only the 19 most common scenes following the work of [8, 20, 21]:

```
[
    'bathroom', 'bedroom', 'classroom', 'computer_room', 'conference_room',
    'corridor', 'dining_area', 'dining_room', 'discussion_area',
    'furniture_store', 'home_office', 'kitchen', 'lab', 'lecture_theatre',
    'library', 'living_room', 'office', 'rest_space', 'study_space'
]
```

To correctly embed the depth images we follow this implementation: https://github.com/facebookresearch/ImageBind/issues/134

D. Retrieval-augmentation visualizations

D.1. Toy example decision boundary

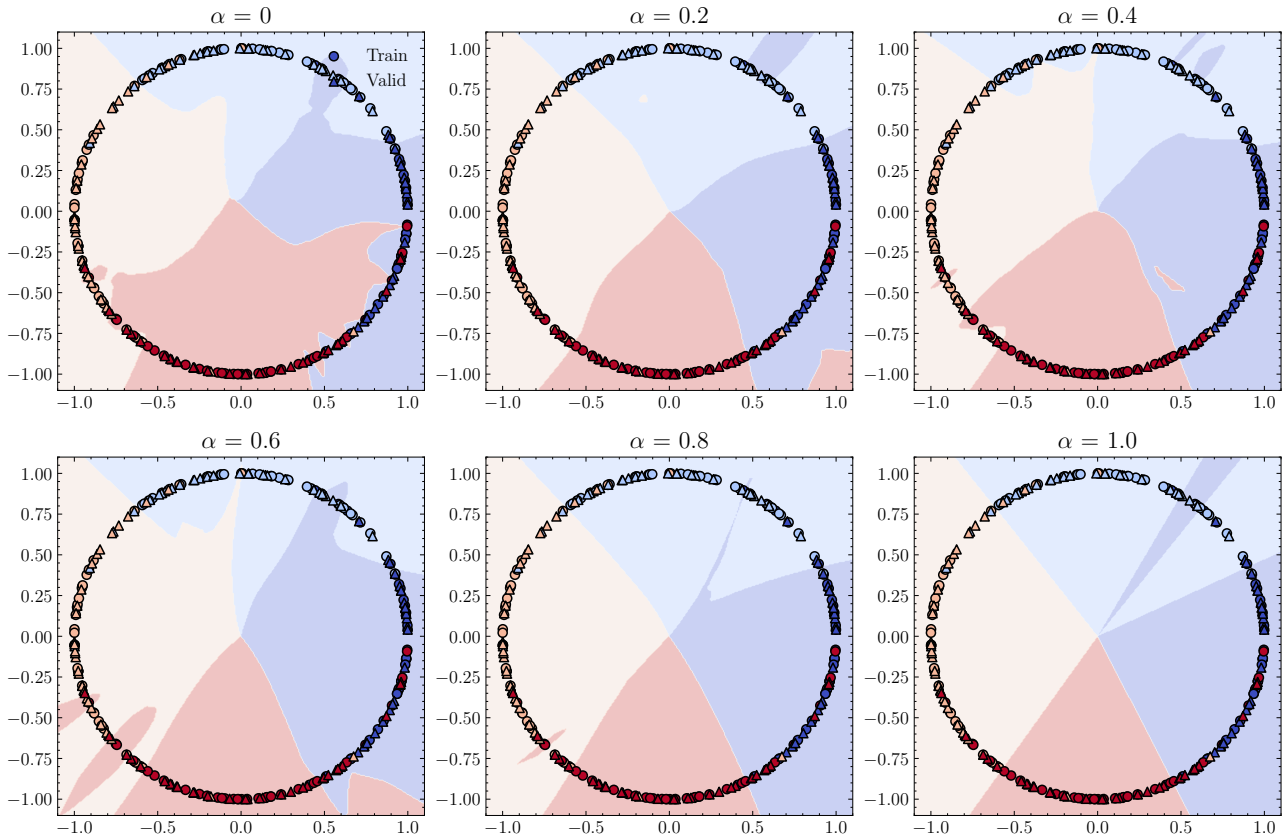


Figure 14. Example of how similar items are retrieved and weighed

Retrieval Augmentation for Adversarial Robust Visual Classification

Supplementary Material

Table 7. Overview of the datasets used with the number of classes at different orders of magnitude.

Dataset	#Training	#Validation	#Classes
SUN RGBD [56]	4845	4659	19

E. Multimodal retrieval

The flexibility of our approach also allows for multi-modal adaptations of the retriever and memory set. We hypothesize that associated information contains a different signal which can denoise the perturbed embedding. Based on the final remark in Section 4.2 we extend the memory to a key-value pair. The similarity search is still done on the key embedding, however, we substitute the key with the coupled value in the weighted embedding mean calculation. To demonstrate the capabilities we use the training split of the SUN RGBD [56] dataset for our memory set, where the key is the RGB embedding and the value the corresponding depth embedding. To do this we use the ImageBind [20] model which is a multi-modal model with a joint embedding space. Note Equation 5 returns the weighted keys, however, now we substitute k_i with another associated value embedding v_i:

$$\hat{e} = \sum_{i=1}^{|M|} w_i v_i, \text{ where } (k_i, v_i) \in M \tag{7}$$

- *SUN RGBD* [56]: This dataset contains almost 10,000 images of indoor scenes. Aside from RGB information it also contains depth information used for other tasks such as object detection and segmentation. We reduce the dataset following previous work [8, 20, 21] to the 19 most frequent classes for classification.
- ImageBind [20]: To demonstrate the multi-modal capabilities of our retrieval-augmented approach (Section 4.2), we select ImageBind. ImageBind is a multi-modal contrastive learning model built on top of the vision transformer architecture. It extends the CLIP model into multiple other modalities such as depth, infrared, and video, but also audio and IMU data. The embedding dimensions for RGB and depth are 1024-dimensional.

We train three models for each modality (image and depth) and in Figure 15 and plot the mean adversarial success rate and the standard error for both. Although the clean accuracy of the model with depth is $60.7\% \pm 0.9$ compared to $66.8\% \pm 1.2$ the adversarial success rate of the model using depth as the returned value is lower.

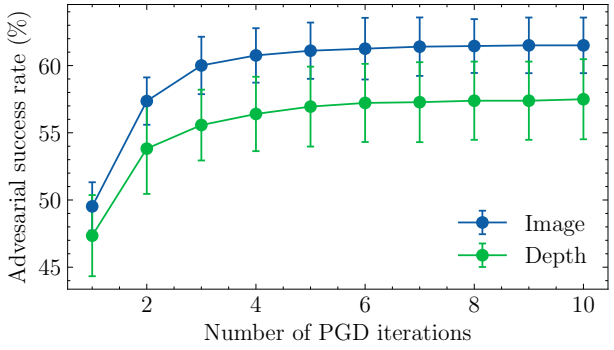


Figure 15. Adversarial success rate for different value modalities on the SUN RGBD dataset for each iteration of the PGD attack.

F. Hard k nearest neighbors

The retrieval pipeline uses a differentiable approximation of the k nearest neighbors algorithm. In this section we use a hard k-NN approach to observe changes. We evaluate a global PGD attack at $\epsilon = 0.05$ and rectangular PGD attack [62] at $\epsilon = 1.0$.

- *ROA (Rectangular Occlusion Attack)* [62]: ROA is a white-box adversarial attack that strategically places rectangular occlusions over input images to degrade model performance. Within each occlusion region, an inner PGD procedure is applied to maximize the model's loss under an L_∞ norm constraint, creating localized, high-impact perturbations. This hybrid approach combines spatial occlusion with gradient-based optimization, making it effective against models that are robust to conventional pixel-wise attacks. ROA is particularly useful for evaluating spatial robustness and models' reliance on localized features.

(a) Clean image (b) Adversarial image at i=30

Figure 16. Rectangular PGD example with a patch size of 30x30 pixels at $\epsilon = 1.0$

Table 8. Overview of hard k-NN retrieval on the SUN RGBD dataset for the rectangular PGD and global PGD attack

Model	Baseline Accuracy (%)	Rectangular PGD			PGD		
		Iteration i			Iteration i		
		i=5	i=15	i=30	i=1	i=10	i=30
ImageBind	74.4	38.5	7.5	1.5	37.6	0.0	0.0
Ours ($\alpha = 0.25$)	73.7	47.2	16.1	4.7	43.1	1.2	0.0
Ours ($\alpha = 0.5$)	73.2	51.3	28.3	14.0	48.3	1.9	0.1
Ours ($\alpha = 0.75$)	72.7	55.1	39.9	30.7	54.4	10.3	8.0
Ours ($\alpha = 1$)	72.3	**59.2**	**49.1**	**45.7**	**58.3**	**25.3**	**18.9**

Additionally we implement a trainable memory attention module [31] as part of the retrieval module. For different values of α with $k = 64$ for the PGD attack and the rectangular PGD, we report the adversarial accuracy at intermediate iterations in Table 8 for image-to-image retrieval. Under a hard k-NN which blocks gradient flow through the retrieval pipeline, the model remains much more robust for more iterations compared to the differentiable model in Table 3.

Again with depth information as the retrieval value, we show how depth information can remain more adversarial robust at different iterations of the rectangular PGD attack in Table 9.

Model	Rectangular PGD (%)					
	RGB $\rightarrow$ RGB			RGB $\rightarrow$ Depth		
	i=5	i=15	i=30	i=5	i=15	i=30
ImageBind	38.5	7.5	1.5	38.5	7.3	1.5
Ours ($\alpha = 0.25$)	47.2	16.1	4.7	48.8	16.0	5.9
Ours ($\alpha = 0.5$)	51.3	28.3	14.0	51.4	30.4	18.5
Ours ($\alpha = 0.75$)	55.1	39.9	30.7	54.9	44.9	37.7
Ours ($\alpha = 1$)	**59.2**	**49.1**	**45.7**	**60.1**	**53.5**	**51.6**

Table 9. RGB vs Depth retrieval for stability against adversarial perturbations

References

[1] Maksym Andriushchenko et al. *Square Attack: a query-efficient black-box adversarial attack via random search*. arXiv:1912.00049 [cs]. July 2020. DOI: 10.48550/arXiv.1912.00049. URL: http://arxiv.org/abs/1912.00049 (visited on 04/15/2025).

[2] Claudine Badue et al. "Self-driving cars: A survey". en. In: *Expert Systems with Applications* 165 (Mar. 2021), p. 113816. ISSN: 09574174. DOI: 10.1016/j.eswa.2020.113816. URL: https://linkinghub.elsevier.com/retrieve/pii/S095741742030628X (visited on 04/29/2025).

[3] Nicholas Carlini and David Wagner. *Towards Evaluating the Robustness of Neural Networks*. arXiv:1608.04644 [cs]. Mar. 2017. DOI: 10.48550/arXiv.1608.04644. URL: http://arxiv.org/abs/1608.04644 (visited on 04/15/2025).

[4] Pin-Yu Chen et al. "ZOO: Zeroth Order Optimization based Black-box Attacks to Deep Neural Networks without Training Substitute Models". In: *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*. arXiv:1708.03999 [stat]. Nov. 2017, pp. 15–26. DOI: 10.1145/3128572.3140448. URL: http://arxiv.org/abs/1708.03999 (visited on 04/15/2025).

[5] Mostafa Dehghani et al. *Scaling Vision Transformers to 22 Billion Parameters*. arXiv:2302.05442 [cs]. Feb. 2023. DOI: 10.48550/arXiv.2302.05442. URL: http://arxiv.org/abs/2302.05442 (visited on 04/28/2025).

[6] Ambra Demontis et al. "Why Do Adversarial Attacks Transfer? Explaining Transferability of Evasion and Poisoning Attacks". en. In: ().

[7] Jia Deng et al. "ImageNet: A large-scale hierarchical image database". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. ISSN: 1063-6919. June 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848. URL: https://ieeexplore.ieee.org/document/5206848/ (visited on 04/16/2025).

[8] Alexey Dosovitskiy et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. arXiv:2010.11929 [cs]. June 2021. DOI: 10.48550/arXiv.2010.11929. URL: http://arxiv.org/abs/2010.11929 (visited on 04/28/2025).

[9] Reuben Feinman et al. *Detecting Adversarial Samples from Artifacts*. arXiv:1703.00410 [stat]. Nov. 2017. DOI: 10.48550/arXiv.1703.00410. URL: http://arxiv.org/abs/1703.00410 (visited on 04/26/2025).

[10] Charlie Giattino et al. "Artificial Intelligence". In: *Our World in Data* (2023).

[11] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. *Explaining and Harnessing Adversarial Examples*. arXiv:1412.6572 [stat]. Mar. 2015. DOI: 10.48550/arXiv.1412.6572. URL: http://arxiv.org/abs/1412.6572 (visited on 04/11/2025).

[12] Alex Graves, Greg Wayne, and Ivo Danihelka. *Neural Turing Machines*. arXiv:1410.5401 [cs]. Dec. 2014. DOI: 10.48550/arXiv.1410.5401. URL: http://arxiv.org/abs/1410.5401 (visited on 01/07/2025).

[13] Kelvin Guu et al. *REALM: Retrieval-Augmented Language Model Pre-Training*. arXiv:2002.08909. Feb. 2020. URL: http://arxiv.org/abs/2002.08909 (visited on 10/15/2024).

[14] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-Term Memory". In: *Neural Computation* 9.8 (1997), pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

[15] Ziniu Hu et al. *REVEAL: Retrieval-Augmented Visual-Language Pre-Training with Multi-Source Multimodal Knowledge Memory*. en. arXiv:2212.05221 [cs]. Apr. 2023. URL: http://arxiv.org/abs/2212.05221 (visited on 10/01/2024).

[16] Mintong Kang et al. *C-RAG: Certified Generation Risks for Retrieval-Augmented Language Models*. arXiv:2402.03181 [cs]. July 2024. DOI: 10.48550/arXiv.2402.03181. URL: http://arxiv.org/abs/2402.03181 (visited on 04/14/2025).

[17] Vladimir Karpukhin et al. *Dense Passage Retrieval for Open-Domain Question Answering*. en. arXiv:2004.04906 [cs]. Sept. 2020. URL: http://arxiv.org/abs/2004.04906 (visited on 09/24/2024).

[18] Paramjit Kaur et al. “Facial-recognition algorithms: A literature review”. In: *Medicine, Science and the Law* 60.2 (2020). _eprint: https://doi.org/10.1177/0025802419893168, pp. 131–139. DOI: 10.1177/0025802419893168. URL: https://doi.org/10.1177/0025802419893168.

[19] Kevin LaGrandeur. “How safe is our reliance on AI, and should we regulate it?” In: *AI and Ethics* 1.2 (May 2021), pp. 93–99. ISSN: 2730-5961. DOI: 10.1007/s43681-020-00010-7. URL: https://doi.org/10.1007/s43681-020-00010-7.

[20] Yann LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998). Publisher: Ieee, pp. 2278–2324.

[21] Patrick Lewis et al. *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. en. arXiv:2005.11401 [cs]. Apr. 2021. URL: http://arxiv.org/abs/2005.11401 (visited on 09/10/2024).

[22] Geert Litjens et al. “A survey on deep learning in medical image analysis”. en. In: *Medical Image Analysis* 42 (Dec. 2017), pp. 60–88. ISSN: 13618415. DOI: 10.1016/j.media.2017.07.005. URL: https://linkinghub.elsevier.com/retrieve/pii/S1361841517301135 (visited on 04/29/2025).

[23] Alexander Long et al. “Retrieval Augmented Classification for Long-Tail Visual Recognition”. en. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA: IEEE, June 2022, pp. 6949–6959. ISBN: 978-1-66546-946-3. DOI: 10.1109/CVPR52688.2022.00683. URL: https://ieeexplore.ieee.org/document/9879687/ (visited on 09/24/2024).

[24] Aleksander Madry et al. *Towards Deep Learning Models Resistant to Adversarial Attacks*. arXiv:1706.06083 [stat]. Sept. 2019. DOI: 10.48550/arXiv.1706.06083. URL: http://arxiv.org/abs/1706.06083 (visited on 03/22/2025).

[25] Maxime Oquab et al. *DINOv2: Learning Robust Visual Features without Supervision*. arXiv:2304.07193. Feb. 2024. URL: http://arxiv.org/abs/2304.07193 (visited on 11/18/2024).

[26] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. *Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples*. arXiv:1605.07277 [cs]. May 2016. DOI: 10.48550/arXiv.1605.07277. URL: http://arxiv.org/abs/1605.07277 (visited on 04/15/2025).

[27] Johannes Stallkamp et al. “The German Traffic Sign Recognition Benchmark: A multi-class classification competition”. en. In: *The 2011 International Joint Conference on Neural Networks*. San Jose, CA, USA: IEEE, July 2011, pp. 1453–1460. ISBN: 978-1-4244-9635-8. DOI: 10.1109/IJCNN.2011.6033395. URL: http://ieeexplore.ieee.org/document/6033395/ (visited on 04/08/2025).

[28] Christian Szegedy et al. *Intriguing properties of neural networks*. arXiv:1312.6199 [cs]. Feb. 2014. DOI: 10.48550/arXiv.1312.6199. URL: http://arxiv.org/abs/1312.6199 (visited on 04/12/2025).

[29] Simon Chi Lok Yu et al. *Evaluating the Adversarial Robustness of Retrieval-Based In-Context Learning for Large Language Models*. arXiv:2405.15984 [cs] version: 1. May 2024. DOI: 10.48550/arXiv.2405.15984. URL: http://arxiv.org/abs/2405.15984 (visited on 04/03/2025).

[30] Djemel Ziou and Salvatore Tabbone. “Edge Detection Techniques-An Overview”. en. In: ().