Semantic Subgraph Extraction Attacks To Convolutional Neural Networks

Gabriele Restuccia [1] **Ilenia Tinnirello** [1] **Francesco Restuccia** [2]

Abstract

Existing adversarial machine learning (AML) methods typically require substantial computational resources. In this work, we investigate how class-level semantic relationships can be exploited to influence adversarial attack performance. Specifically, we first aim to quantify the effect of known factors, such as semantic similarity–defined as the degree of shared intrinsic attributes–on attack efficiency. Experiments on CIFAR-100 and Tiny-ImageNet using VGG and ResNet architectures show that targeting semantically similar classes can reduce perturbation magnitudes and iterations by up to 41% and 23%, respectively. Motivated by these findings, we introduce a lightweight, one-shot, semantic subgraph extraction attack (SSEA), which constructs semantic subgraphs by leveraging class-level semantic relationships between source and adversarial target classes. Our method extracts subgraphs in a single inference pass, requires no fine-tuning or external models, preserves the original network weights, and integrates seamlessly into any white-box attack scenario. On CIFAR-100, SSEA improves the Top-1 attack success rate (ASR) by up to 8.17% for PGD on VGG-19 and nearly doubles the effectiveness of the Jitter attack. Additionally, our approach reduces floating-point operations and model size by up to 18% and 42%, respectively.

1. Introduction

Despite the growing attention toward Vision Transformers (ViTs) (Khan et al., 2023; Han et al., 2022), convolutional neural networks (CNNs) still offer greater advantages in specific computer vision scenarios such as resource and data-constrained real-time applications in mobile and embedded vision systems, where low floating point operations (FLOPs) (Vasu et al., 2023; Li et al., 2021) and high computational efficiency (Khan et al., 2023) are essential. Moreover, modern CNNs achieve accuracy comparable to ViTs (Todi et al., 2023), underscoring their continued relevance in contemporary research. However, CNNs are also known to be susceptible to adversarial perturbations (Li et al., 2019; Che et al., 2021; Wang et al., 2023). Existing adversarial machine learning (AML) often requires significant computational resources, which strongly depend on the given CNN, dataset size, model complexity and performance goals (Hussain et al., 2022; Ayaz et al., 2023). For example, while advanced AML such as Carlini & Wagner (C&W) (Carlini & Wagner, 2017) can reach up to 100% effectiveness (Zhang & Wu, 2020), it can be up to 13,000 times slower (Harry24k, 2024) than simpler attacks like Projected Gradient Descent (PGD) (Madry et al., 2017) and Fast Gradient Sign Method (FGSM) (Goodfellow et al., 2014). *This computational overhead highlights a gap between existing AML work and their real-world applicability.* This challenge becomes critical when CNNs are updated periodically over time, for example, with fine-tuning (Ribani & Marengoni, 2019), few-shot learning (Wang et al., 2020) and test-time adaptation strategies (Liang et al., 2024).

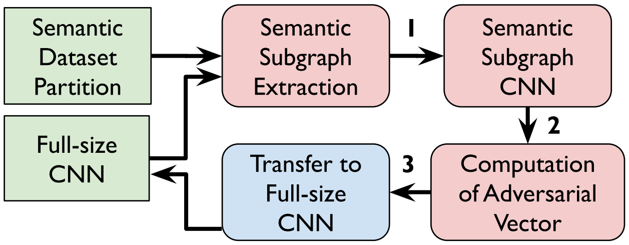

Semantic Dataset Partition
Full-size CNN
Semantic Subgraph Extraction
1
Semantic Subgraph CNN
2
Computation of Adversarial Vector
3
Transfer to Full-size CNN

Figure 1: Overview of SSEA. We first extract a subgraph CNN from the full-size CNN evaluating a semantic cluster of the dataset (1), we execute an existing white-box attack on the subgraph (2), and we transfer the adversarial vector to the full-size CNN (3).

To address these limitations, we propose a novel approach to AML named *semantic subgraph extraction attack (SSEA)*. Our method exploits the inherent capacity of CNNs to capture semantic correlations among spatial features in multidimensional data (Talaei Khoei et al., 2023). Specifically, we investigate whether leveraging semantic relationships

[1] University of Palermo, Italy [2] Northeastern University, MA, USA. Correspondence to: Gabriele Restuccia <gabriele.restuccia@unipa.it>, Ilenia Tinnirello <ilenia.tinnirello@unipa.it>, Francesco Restuccia <f.restuccia@northeastern.edu>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025. Copyright 2025 by the author(s).

between classes can improve the efficiency of adversarial attacks.

We operate within a white-box adversarial threat model (Chakraborty et al., 2021), which assumes full access to the target model's parameters and architecture. This setting is critical for exposing vulnerabilities and guiding the development of robust defenses, including those applicable to black-box scenarios. Moreover, white-box attacks are practically relevant in contexts where models trained locally are later deployed through shared platforms, thereby exposing them to potential insider threats (Santos et al., 2025; Bonati et al., 2021).

Prior studies have suggested that certain classes may inherently be more susceptible to targeted adversarial perturbations (Goodfellow et al., 2014; Halmosi et al., 2024; Carlini & Wagner, 2017). Motivated by this, we conduct experiments to quantitatively assess how class semantics influence adversarial vulnerability in widely-used CNNs architectures, including ResNet-50, VGG-16, and VGG-19. Applying the C&W L_2 attack on established datasets – CIFAR-100 and selected subsets of Tiny-ImageNet (Li et al., 2024) – we measure the effect of semantic similarity, defined as the degree of shared attributes or meaning among classes. Our findings indicate that adversarial attacks targeting semantically similar classes significantly reduce input perturbation magnitudes by up to 41% on CIFAR-100 and 38% on Tiny-ImageNet, while concurrently decreasing the number of required attack iterations by up to 23% and 10%, respectively.

Following on these results, we further incorporate the insights from (Sayyed et al., 2023), which demonstrate that full-scale CNNs can be effectively reduced to smaller surrogate networks focused on semantically defined class subsets. Rather than operating on entire networks, our SSEA is specifically designed to exploit these reduced networks, referred to as *subgraphs*. Importantly, SSEA does not require external models or incremental fine-tuning. Instead, subgraph extraction is performed via a single inference pass on a targeted data subset encompassing coarse-grained categories of both the original and adversarial target classes. This one-shot, on-the-fly extraction preserves the original weights and network configuration.

The core mechanism of our approach involves generating adversarial inputs using these reduced-complexity subgraphs and subsequently transferring the perturbations to the original, full-scale CNNs (Figure 1). This strategy significantly reduces computational complexity without degrading attack effectiveness. Specifically, SSEA lowers the computational burden by reducing FLOPs and model parameters by up to 18% and 42%, respectively, compared to attacks performed directly on full-scale models.

To evaluate the efficacy of our subgraph-based attacks, we compare adversarial inputs crafted using subgraphs against those generated directly on the full-scale models. Notably, adversarial examples generated on semantic subgraphs frequently surpass full-model counterparts in terms of attack success rate (ASR). For instance, on CIFAR-100, subgraph-derived attacks consistently yield superior results across various attack types, with VGG-19 subgraphs attaining an 8.17% higher relative Top-1 ASR for PGD attacks and nearly doubling the effectiveness of Jitter attacks (30.91% versus 15.83%). Although some configurations exhibit marginal performance reductions, the overall effectiveness of the Jitter attack improves consistently across all architectures and datasets; specifically, ResNet-50 subgraphs demonstrate nearly twice the effectiveness compared to their full-model counterparts on Tiny-ImageNet (13.23% vs. 6.67%). Our SSEA methodology introduces negligible additional latency and offers new insights into semantic interactions within CNNs, thereby presenting a robust and efficient enhancement for AML performance.

2. Background and Intuition

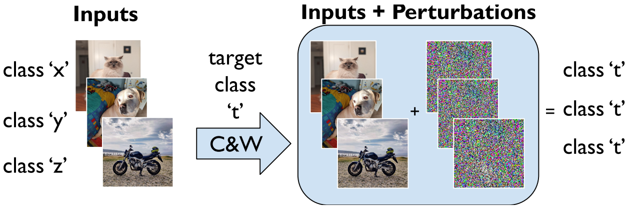


Figure 2: Targeted C&W attack on images from classes 'x', 'y', and 'z'. Specific perturbations are added to each image, creating adversarial examples that are all misclassified as the target class 't'.

2.1. Adversarial Attacks on Neural Networks

An *adversarial input* for a model is crafted by intentionally altering an input to shift its representation into an incorrect classification region within the model's decision space (Huang et al., 2019). This alteration involves modifying the input data x with a noise vector δ, termed *perturbation*, to induce errors in neural network decisions. These adversarial inputs, often constrained by a maximum distance, employ various distance metrics to measure perturbation effectiveness–such as sparsity (L_0), total extent (L_1), energy (L_2), and maximum change (L_∞) (Papernot et al., 2016c;b; Chen et al., 2018; Carlini & Wagner, 2017; Goodfellow et al., 2014)–and are categorized into *targeted* and *untargeted* attacks. Targeted attacks aim to misclassify inputs into a specific class, while untargeted attacks simply aim to cause any misclassification.

Using RGB images, the input data x is a three-dimensional

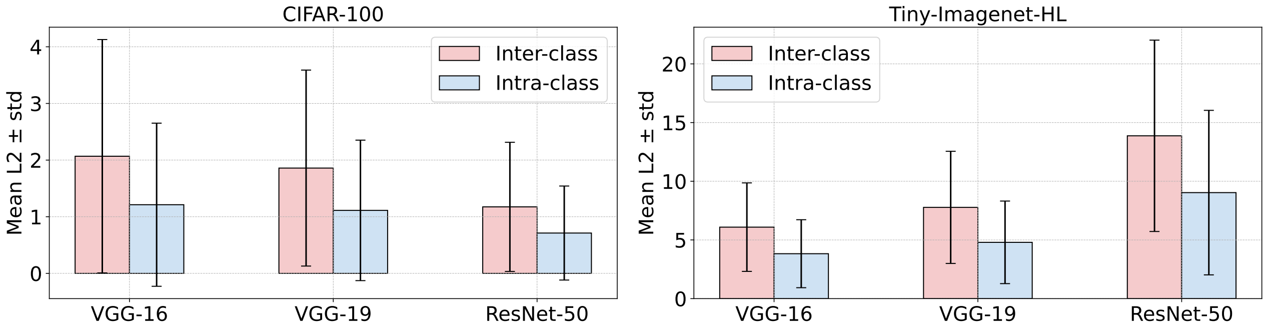


Figure 3: Adversarial attack mean±std L_2 perturbation magnitude comparison. Lower values mean less perturbation to the original input.

matrix representing color channels in an image of size $3 \times h \times w$. Figure 2 shows how adversarial perturbations, unique to each image class x, y, and z, result in misclassification as class t. We employ the C&W L_2 attack (Carlini & Wagner, 2017), aiming to minimize the perturbation δ while ensuring $C(x + \delta) = t$:

$$\begin{aligned} &\min_{\delta} ||\delta||_p + c \cdot f(x + \delta) \\ &\text{s.t.} \quad x + \delta \in [0, 1]^{3 \cdot h \cdot w} \end{aligned} \tag{1}$$

Here, $||\delta||_p$ is the L_2 norm of the perturbation, measuring its magnitude. The function $f(x + \delta)$ is often implemented via cross-entropy loss, where $f(x+\delta) \leq 0$ implies $C(x+\delta) = t$. The scalar c balances the norm of the perturbation and the classification objective. This attack utilizes iterative gradient descent, adjusting c based on whether $C(x + \delta_c) = t$, iterating up to $n_g \cdot n_c$ times, where n_c corresponds to binary search steps and n_g to iterations for each binary search step. The smallest δ achieving $C(x + \delta_c) = t$ is selected as the final perturbation; if none meet this criterion, the attack fails.

2.2. Related work

A limited number of studies have explored the manipulation of CNN architectures to accelerate attacks, reduce model complexity, or utilize surrogate models for faster execution. (Qin et al., 2023) introduce a Meta-Transfer Attack (MTA) framework that employs a Meta-Surrogate Model (MSM) to generate highly transferable adversarial examples. The MSM is optimized to enhance the transferability of these examples to various target models, thereby improving the overall effectiveness of the attack. However, this approach requires extensive trial-and-error and fine-tuning, primarily targeting black-box scenarios- (Du et al., 2021) present ”Fast C&W,” a rapid adversarial attack algorithm using a deep encoder network to efficiently generate adversarial examples for SAR target recognition systems. While this method significantly improves attack speed and maintains high effectiveness, it is confined to the C&W attack and does not generalize to other input types or tasks. Similarly, (Wu et al., 2020) explore the manipulation of skip connections to enhance the transferability of adversarial inputs between different models, but they do not address the acceleration or simplification of the attack process. (Huang et al., 2019) focus on refining existing adversarial examples to improve transferability across different models, introducing a fine-tuning step but not emphasizing the speed of the attack process. Additionally, works by (Yao et al., 2019), (Matyasko & Chau, 2021), and (Zhang et al., 2020) propose methods for creating faster attacks or optimizing existing ones. However, these studies typically concentrate on a single type of attack and do not explore the optimization of multiple attacks or the reduction of model complexity. We also consider generation-based adversarial attacks, where a generative model is trained offline to produce perturbations that mislead target models at inference time without iterative optimization (Poursaeed et al., 2018; Wang et al., 2022; Baytaş & Deb, 2023). Nevertheless, such approaches might require substantial offline training, architectural tuning, and access to auxiliary data, making them less generalizable and more complex to deploy than on-the-fly iterative attacks. Finally, various neural network manipulation techniques have been proposed as defense mechanisms (Sehwag et al., 2020; Wu & Wang, 2021; Ye et al., 2019; Dhillon et al., 2018; Jordao & Pedrini, 2021; Lin et al., 2019; Vemparala et al., 2021). However, their potential to enhance the efficiency of attacks has not been investigated.

2.3. Intuition on the impact of semantic similarity on the attack performance

”Semantic similarity” describes the degree to which two classes share attributes, meanings, or contexts. Classes are considered semantically similar if they possess related meanings or contexts, a concept applicable not only in image classification but also in other domains such as natural

language processing (Mikolov et al., 2013). Findings from (Sayyed et al., 2023) reveal that semantically similar inputs activate a greater number of common filters, particularly in the early layers of a CNN. For example, images of birds such as sparrows and eagles tend to activate more similar filters than those of birds and automobiles. In addition, in some settings CNN inputs might come only from a limited set of classes which are usually semantically similar and are highly correlated over time (Sayyed et al., 2023).

Based on this definition, we aim to quantify the extent to which adversarial attacks targeting semantically similar classes–specifically, intra-class attacks between fine-grained classes within the same coarse category–can reduce perturbation magnitude and improve attack performance, compared to inter-class attacks spanning different coarse categories.

We measure the convergence time of a successful C&W attack by the total iterations $N(x) \leq n_g \cdot n_c$ needed to find the optimal perturbation. We evaluate VGG-16, VGG-19 (Simonyan & Zisserman, 2014), and ResNet-50 (He et al., 2016) CNNs using CIFAR-100 (Krizhevsky et al., 2009) and Tiny-Imagenet (Chrabaszcz et al., 2017). These models and datasets, though not the most recent, are among the most popular and comprehensive. They are widely used in machine learning (ML) studies (Balderas et al., 2023) and serve as a well-known baseline for our proposed work. CIFAR-100 consists of 100 *fine* classes grouped into 20 *coarse* classes, each containing five semantically similar fine classes. For instance, the "aquatic mammals" superclass includes fine classes like "beaver," "dolphin," "otter," "seal," and "whale." In contrast, Tiny-Imagenet does not inherently support fine and coarse class divisions. Therefore, we grouped 25 of Tiny-Imagenet's fine classes into 5 coarse classes, each containing 5 fine classes based on semantic and visual similarities. We refer to this modified dataset as "Tiny-HL" (Hierarchical Labeling) to distinguish it from the original. Tiny-HL images were upscaled to 224x224 pixels to match the original Imagenet dataset, which was excluded from this evaluation due to hardware limitations and time constraints. Implementation procedures and reproducibility guidelines for this hierarchical grouping are provided in our code at (Restuccia, 2025).

To evaluate attack properties consistently, we filtered the dataset D by keeping only the images that were originally correctly predicted. All subsequent evaluations in this paper focus exclusively on these correctly predicted images.

We perform multiple attacks on each image in our dataset, transitioning each input image from its original fine label to every possible target fine class within the datasets. We configured the C&W attack with a learning rate of 5×10^{-3}, a confidence level of 0, and a bound on the per-pixel perturbation, considering the range of the original datasets after standard scaling. The attack strategy included $n_c = 5$ binary search steps and $n_g = 1000$ iterations.

Table 1 presents the average number of total iterations $E_{D'}[N]$ required to find the optimal adversarial sample for all images in D', while Figure 4 details the mean±std L_2 perturbations across datasets and models. A consistent pattern emerges: intra-class attacks demonstrate significantly greater efficiency than inter-class attacks, evident in both convergence time and perturbation size. For Tiny-HL, results show up to 9.79% faster mean convergence for intra-class attacks across models, with reductions in the mean L_2 perturbation sizes of up to 38.32%. These findings are consistent with those for CIFAR-100, where intra-class attacks achieved up to 22.88% faster mean convergence and a reduction in mean perturbation size of up to 50.85%. Figure 4 illustrates two subsets of adversarial attacks on the ResNet model using the CIFAR-100 dataset. These subsets involve images that shift from their original fine class to a target fine class in a different coarse class. We specifically target fine label 82 within coarse label 2 and fine label 35 within coarse label 14. The y-axis represents the perturbation magnitude, while the x-axis indicates the original coarse class of the images. We calculate the median perturbation magnitude for the sample set. Our analysis, shown by the blue trajectory, indicates that images from coarse class 14, which matches the target class's coarse class, have lower perturbation magnitudes. This subset also shows the smallest median perturbation among attacks where the original and target coarse classes align. However, this trend is not universal. The green trajectory highlights scenarios where minimal perturbation magnitudes occur across different source and target coarse classes, underscoring the complex dynamics of adversarial perturbations.

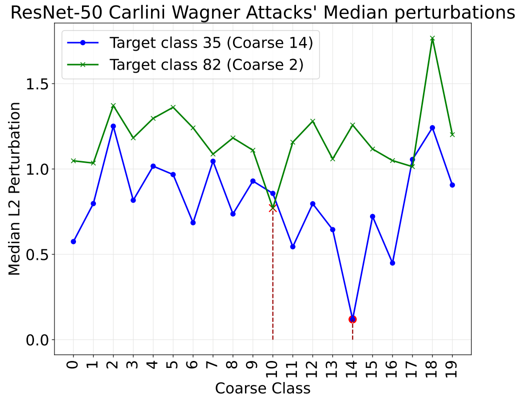


Figure 4: ResNet-50 easiest and hardest target attack between coarse classes. Lower values mean less perturbation to the original input.

Table 1: Comparison of Adversarial Attack Efficiency on CIFAR-100 and Tiny-HL. Bold values indicate improved performance (lower iterations and higher speed-up) with intra-class attacks.

	CIFAR-100			Tiny-HL		
CNN	**Inter** ($E_{D'}[N]$)	**Intra** ($E_{D'}[N]$)	**% Faster**	**Inter** ($E_{D'}[N]$)	**Intra** ($E_{D'}[N]$)	**% Faster**
VGG-16	2037.78	**1571.44**	**22.88%**	2365.98	**2197.45**	**7.12%**
VGG-19	2010.13	**1642.53**	**18.29%**	2558.09	**2348.83**	**8.18%**
ResNet-50	1617.12	**1283.58**	**20.63%**	2858.18	**2578.30**	**9.79%**

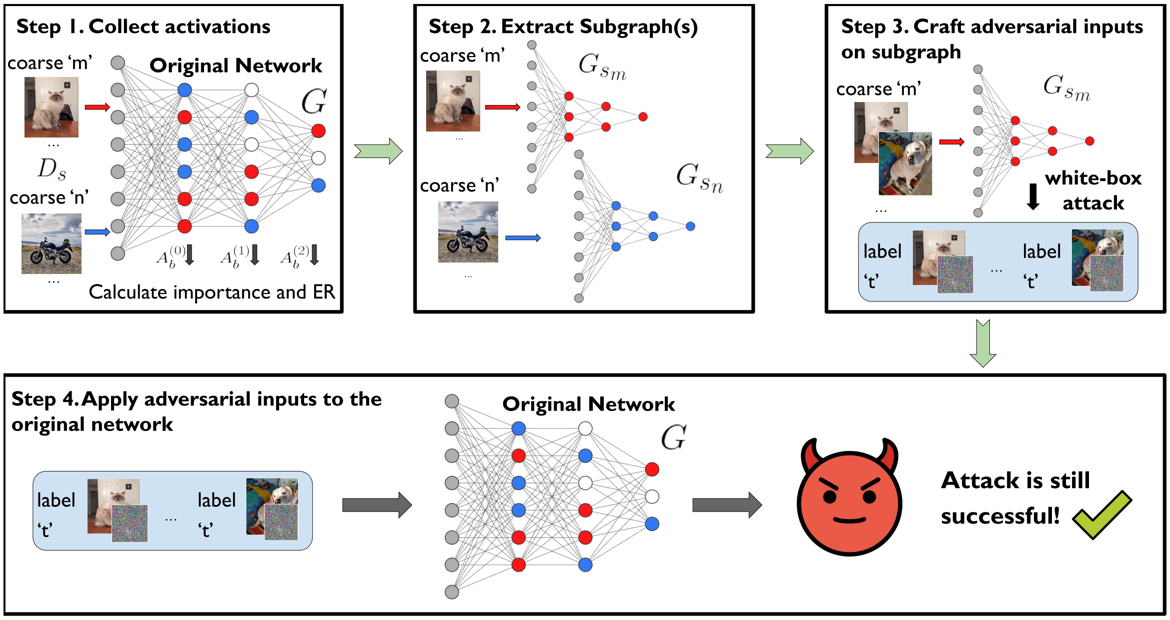


Figure 5: Simplified representation of all SSEA operations.

3. Our approach

We build on the insights from previous analyses and the intuition introduced by (Sayyed et al., 2023), who demonstrate that semantically similar classes can be effectively clustered to extract compact, class-specific models from full-sized ones. Following a similar philosophy, our SSEA targets CNNs by focusing on a constrained subset of semantically related classes, typically drawn from one or two coarse categories, and extracts corresponding subgraphs.

This extraction process resembles structured pruning, particularly activation-based methods (He & Xiao, 2023), though we introduce no novel pruning technique. Instead, we readapt and implement the DropNet methodology (Tan & Motani, 2020), which selects filters based on the average magnitude of post-activation feature maps, and incorporate this criterion into our attack framework. Unlike conventional structured pruning techniques (He & Xiao, 2023), as well as DropNet itself, our implementation requires no fine-tuning and preserves the original network weights. As a result, it retains most of the full model's receptive field and saliency maps. Consequently, we modify the CNN architecture according to the input and adversarial target classes rather than optimizing the attacks directly. This approach minimizes deviations from the original attack direction when applied to the full-size model. Our objective is to maintain the effectiveness of adversarial inputs when transferred back to the original model, as significant directional changes can reduce attack potency (Huang et al., 2019). A further key advantage is that subgraph extraction is performed at runtime and on-the-fly in a single pass, making it compatible with any white-box model and attack scenario without introducing significant latency.

3.1. Understanding SSEA's process

As illustrated in Figure 5 and detailed in Algorithm 1, our subgraph extraction process begins by performing inference on a dataset subset, D_s, which includes samples from all fine-grained classes within the targeted and source coarse classes. We define G as the function representing the CNN's graph structure, with weights W. As in the original DropNet (Tan & Motani, 2020), for each convolutional layer l and batch b, let $A_b^{(l)} \in \mathbb{R}^{N_b \times C_l \times H_l \times W_l}$ be the ReLU outputs.

Algorithm 1 Semantic Subgraph Extraction Attack (SSEA)

```
Input: Network $G$, Dataset subset $D_s$, Extraction ratios $\mathbf{ER}$, Target class $t$
Output: Extracted subnetwork $G_s$, Adversarial inputs $X_{\text{adv}}$
Initialize activation statistics for all layers
// Collect activation statistics
for each batch $b$ in $D_s$ do
    Forward pass through $G$
    for each convolutional layer $l$ do
        Compute channel importance based on ReLU activations
        Update running statistics for layer $l$
    end for
end for
// Extract subgraph
for each layer $l$ do
    Retain top $(1 - ER_l)$ fraction of channels based on importance
end for
Adjust output layer to match target classes
// Generate adversarial examples
Craft adversarial inputs $X_{\text{adv}}$ using $G_s$ targeting class $t$
Transfer $X_{\text{adv}}$ to the full network $G$
return $G_s$, $X_{\text{adv}}$
```

We compute the mean absolute activation per channel c:

$$M_{b,c}^{(l)} = \frac{1}{N_b\, H_l\, W_l} \sum_{n=1}^{N_b} \sum_{h=1}^{H_l} \sum_{w=1}^{W_l} \left| A_b^{(l)}[n,c,h,w] \right|. \qquad (2)$$

A running average $\bar{M}_c^{(l)}$ is updated across batches as $\bar{M}_c^{(l)} \leftarrow \frac{T \cdot \bar{M}_c^{(l)} + M_{b,c}^{(l)}}{T + N_b}$, where T tracks the total samples processed.

This channel importance scoring mechanism forms the core of our subgraph extraction algorithm, identifying the most salient channels while removing less relevant ones.

The extracted subgraph's output layer is adjusted to match the targeted coarse classes. Ultimately, we obtain a subgraph G_s, a compact version of G with a subset of its weights W_s, such that:

$$G_s(W_s) = E(G(W), D_s, ER) \qquad (3)$$

The extraction ratio (ER) represents the proportion of channels to be pruned at each layer. For example, an ER of 0.3 for a specific layer means that 30% of its channels are removed, retaining the top 70% most important channels based on activation statistics. Different ERs can be applied to different layers, allowing for non-uniform pruning across the network.

The weights $W_s \subset W$ maintain identical values to their corresponding weights in the original network, ensuring the preservation of learned representations in G_s. This subgraph is used to generate adversarial inputs, X_{adv}, utilizing any white-box attack method. These inputs are then transferred to the original full-size CNN, effectively simulating an attack crafted directly on the original network. Our goal is to ensure that X_{adv} remains effective, meaning that the original full-size CNN still misclassifies the adversarial inputs as the target class t. The effectiveness of the attack is confirmed if the output of f from G satisfies $f(X_{\text{adv}}) = t$.

4. Experimental results

4.1. Setup

We implement our experiments using the PyTorch deep learning framework (Paszke et al., 2019). Adversarial attacks utilized in our evaluation include FGSM (Goodfellow et al., 2014), PGD (Madry et al., 2017), MIFGSM (Dong et al., 2018), and Jitter (Schwinn et al., 2023), provided by the TorchAttacks library (Kim, 2020). Additionally, the Carlini-Wagner (CW) attack from the CleverHans library (Papernot et al., 2016a) is employed exclusively in the first evaluation of semantic similarity, due to its computational intensity making it unsuitable for extensive comparisons in ASR assessments. All codebase, along with the trained model weights, is open-source and made publicly accessible and can be found at (Restuccia, 2025).

4.2. Comparison of CNN Complexity Reduction

We assess the computational cost of crafting adversarial inputs in terms of FLOPs, which are determined by the size, architecture, and complexity of the activation functions in CNNs. This metric provides a hardware-independent evaluation of model complexity (Chen et al., 2023) and its impact on memory usage. Specifically, FLOPs quantify the operations required for a single forward pass through the model. In the context of AML, they represent the computational operations needed for multiple iterations of forward and backward passes to generate adversarial examples. It is important to note that reducing model parameters does not necessarily result in a proportional reduction in FLOPs; the effect depends on the specific architecture and layer complexity.

To determine the optimal parameter reduction configuration for each model, we conducted a systematic evaluation of different ERs. We prioritized configurations that transform the CNNs into a ”funnel” structure, with more aggressive reduction in the later layers. For each architecture, we selected the configuration that maintained an average top-1 accuracy of at least 98% across all test batches while maximizing parameter reduction.

Figure 6 demonstrates the accuracy-parameter tradeoff for VGG-16 (Figure 6a), VGG-19 (Figure 6b), and ResNet-50 (Figure 6c) architectures. The plots reveal how model performance gradually degrades as more parameters are removed. Notably, we observe that sudden descending peaks

Table 2: Comparison of FLOPs (in millions) and Parameter Counts (in millions) Between Original Models and Their Extracted Subgraphs

CNN	Dataset	FLOPs (Original → Subgraph)	Parameters (Original → Subgraph)
VGG-16	CIFAR-100	314.35 → 271.22 (**-13.72%**)	14.77 → 8.82 (**-40.28%**)
VGG-16	Tiny-HL	314.41 → 309.04 (**-1.71%**)	14.83 → 13.41 (**-9.55%**)
VGG-19	CIFAR-100	399.39 → 328.46 (**-17.76%**)	20.08 → 11.63 (**-42.08%**)
VGG-19	Tiny-HL	399.45 → 394.08 (**-1.34%**)	20.14 → 18.72 (**-7.03%**)
ResNet-50	CIFAR-100	1311.78 → 1170.34 (**-10.78%**)	23.71 → 19.12 (**-19.33%**)
ResNet-50	Tiny-HL	84.73 → 83.28 (**-1.71%**)	23.92 → 22.47 (**-6.04%**)

Table 3: Comparison of subgraph and full CNNs on CIFAR-100 ($\epsilon = 0.1$, 20 steps) and Tiny-HL ($\epsilon = 0.01$, 5 steps). Attack success rates (%) under various attacks. In the Subgraph rows, bold numbers indicate superior performance over the Full approach.

CNN	Approach	ASR Top-k	CIFAR-100				Tiny-HL			
			PGD	FGSM	MIFGSM	Jitter	PGD	FGSM	MIFGSM	Jitter
VGG-16	Subgraph	Top-1	**72.53%**	**2.49%**	56.06%	**29.48%**	84.58%	30.56%	81.85%	**20.72%**
		Top-3	**80.22%**	**6.35%**	66.62%	**44.18%**	96.17%	46.94%	93.84%	**29.02%**
		Top-5	**83.65%**	**9.58%**	71.79%	**50.37%**	98.29%	54.44%	96.46%	**32.68%**
	Full	Top-1	71.64%	2.28%	62.95%	15.29%	99.20%	36.89%	97.80%	13.85%
		Top-3	76.27%	6.07%	69.14%	17.48%	99.84%	51.88%	99.16%	16.38%
		Top-5	80.68%	9.21%	75.17%	18.62%	99.89%	59.06%	99.47%	17.86%
VGG-19	Subgraph	Top-1	**72.79%**	**2.50%**	56.42%	**30.91%**	76.26%	21.52%	69.60%	**18.10%**
		Top-3	**83.01%**	**6.75%**	**69.56%**	**45.55%**	91.04%	36.09%	85.72%	**26.45%**
		Top-5	**86.70%**	**10.22%**	**75.27%**	**51.52%**	94.36%	43.17%	89.75%	**30.13%**
	Full	Top-1	67.29%	2.31%	59.53%	15.83%	95.87%	26.73%	91.14%	13.49%
		Top-3	72.88%	6.58%	67.07%	18.17%	98.42%	41.45%	96.08%	16.24%
		Top-5	77.28%	10.11%	72.39%	19.34%	98.88%	47.95%	97.28%	17.74%
ResNet-50	Subgraph	Top-1	80.68%	**2.27%**	62.47%	**24.79%**	56.33%	9.17%	44.92%	**13.23%**
		Top-3	88.87%	**6.11%**	74.87%	**34.40%**	72.57%	18.55%	61.08%	**21.49%**
		Top-5	91.66%	**9.59%**	79.97%	**37.96%**	78.64%	24.35%	68.23%	**25.68%**
	Full	Top-1	93.06%	2.13%	81.93%	16.72%	73.71%	9.93%	64.70%	6.67%
		Top-3	95.10%	5.68%	86.45%	17.83%	82.26%	18.61%	74.18%	9.38%
		Top-5	96.24%	8.94%	89.00%	18.54%	86.16%	24.75%	78.87%	11.38%

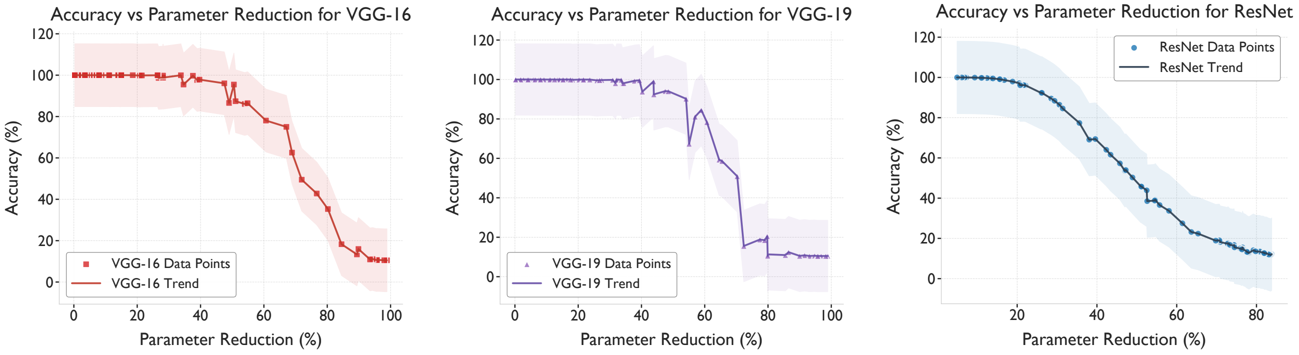
in accuracy depend not only on the quantity of parameters removed but also on which specific layers are manipulated—a phenomenon particularly evident in Figure 6b. Our experiments confirmed that optimal reduction follows a funnel-like pattern. This systematic approach enabled us to identify optimal subgraph configurations that balance performance preservation with computational efficiency.

Table 2 summarizes the computational efficiency gains achieved through our subgraph extraction approach. On CIFAR-100, we achieved substantial parameter reductions of 40.28% and 42.08% for VGG-16 and VGG-19 respectively, alongside FLOPs reductions of 13.72% and 17.76%. ResNet-50 showed a notable 19.33% parameter reduction with a 10.78% FLOPs decrease. The Tiny-HL dataset yielded more modest improvements, with parameter reductions between 6.04-9.55% and minimal FLOPs savings (1.34-1.71%). Even these smaller reductions are valuable for memory-intensive deep CNNs (Han et al., 2015), potentially easing deployment constraints on memory-limited devices.

4.3. SSEA ASR Evaluation after Transfer

We evaluate SSEA using the ASR as our success metric. ASR measures the percentage of adversarial attacks that successfully mislead a neural network into making incorrect predictions. Additionally, the Top$_k$ ASR represents the percentage of adversarial attacks that successfully exclude the true label from the model's Top$_k$ predictions, instead predicting the target adversarial class. In this evaluation, we generate adversarial inputs using both the subgraph-extracted CNN and the full-size CNN. We then compare the full-size CNN's predictions using adversarial inputs from the subgraphs with those crafted directly on the full-size CNN. The ASR for both sets of inputs is measured on the full-size CNN.

In the context of AML, transferability refers to the ability



(a) VGG-16

(b) VGG-19

(c) ResNet-50

Figure 6: Accuracy vs. Parameter Reduction for different CNN architectures. The plots illustrate how model performance changes as parameters are reduced.

of adversarial inputs crafted for one model to deceive other models. This occurs when adversarial inputs cause misclassification in a victim model with a targeted adversarial class. In our study, transferability is localized, as the source and target models share the same weights. Typically, a decrease in ASR is expected when transferring adversarial inputs to a different model. However, Table 3 reveals a remarkable finding: despite the substantial parameter and computational reductions shown in Table 2, our subgraph approach not only maintains but often enhances attack effectiveness. On CIFAR-100, VGG-16 and VGG-19 subgraphs consistently outperform their full counterparts across PGD, FGSM, and Jitter attacks, with VGG-19 subgraphs achieving up to 5.5 percentage points increase, corresponding to 8.17% relative improvement, of Top-1 ASR for PGD and nearly double the effectiveness for Jitter attacks (30.91% vs. 15.83%). While ResNet-50's subgraph shows modest performance decreases for some attacks on CIFAR-100, it still demonstrates significantly improved Jitter attack performance. On Tiny-HL, although full models generally maintain higher ASRs for PGD, FGSM, and MIFGSM, our subgraph approach consistently outperforms full models on Jitter attacks across all architectures, with ResNet-50 showing a particularly impressive nearly 2× improvement (13.23% vs. 6.67%). These results confirm that our targeted subgraph extraction not only delivers substantial computational and memory efficiency but also preserves–and in many cases enhances–adversarial attack potency. Notably, Jitter attacks consistently benefit from our subgraph approach across all architectures and datasets, suggesting a potential structural advantage that merits further investigation in future work.

4.4. Conclusions and Future Work

This paper introduces SSEA, a novel approach that addresses the computational challenges of AML while maintaining, and often enhancing, attack effectiveness. By targeting semantically correlated features and extracting optimized subgraphs from CNNs, our method reduces computational complexity without requiring fine-tuning or additional training. Our comprehensive evaluation demonstrates that SSEA achieves substantial efficiency gains, reducing parameters by up to 42% and FLOPs by up to 18% across various model architectures. Importantly, these improvements do not come at the expense of attack performance; adversarial examples generated on semantic subgraphs frequently outperform those crafted on full models. This effect is particularly pronounced in Jitter attacks, where effectiveness nearly doubles in some scenarios.

A natural question that arises is why subgraph-based attacks can exhibit superior performance. We hypothesize that pruning irrelevant filters may reduce gradient noise, thereby improving alignment with discriminative features. In the specific case of noise-based attacks such as Jitter, constraining the perturbation path to semantically relevant channels may concentrate energy along salient directions, amplifying the attack's effectiveness. These hypotheses remain speculative and require dedicated empirical investigation in future work. Further extensions of this research include exploring the potential of SSEA as a defensive mechanism, for example through adversarial training. Also, while this work focuses on CNNs, extending these techniques to transformer-based architectures is an open challenge that requires dedicated investigation. Given the fundamental architectural and operational differences between CNNs and ViTs, such an extension is non-trivial and beyond the scope of this study. The structural insights gained through SSEA may also contribute to the development of more robust network architectures inherently resistant to adversarial manipulation.

Acknowledgement

This work has been funded in part by the National Science Foundation under grant CNS-2312875, by the Air Force Office of Scientific Research under contract number FA9550-23-1-0261, by the Office of Naval Research under award number N00014-23-1-2221.

This work has also been partly supported by the ISP5G+ project (Innovative Security Paradigms for beyond 5G), funded under the Italian National Recovery and Resilience Plan (PNRR), through the SERICS partnership.

References

Ayaz, F., Zakariyya, I., Cano, J., Keoh, S. L., Singer, J., Pau, D., and Kharbouche-Harrari, M. Improving robustness against adversarial attacks with deeply quantized neural networks. *arXiv preprint arXiv:2304.12829*, 2023.

Balderas, L., Lastra, M., and Benítez, J. M. Optimizing convolutional neural network architecture. *arXiv preprint arXiv:2401.01361*, 2023.

Baytaş, İ. M. and Deb, D. Robustness-via-synthesis: Robust training with generative adversarial perturbations. *Neurocomputing*, 516:49–60, 2023.

Bonati, L., D'Oro, S., Polese, M., Basagni, S., and Melodia, T. Intelligence and learning in o-ran for data-driven nextg cellular networks. *IEEE Communications Magazine*, 59 (10):21–27, 2021.

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pp. 39–57. Ieee, 2017.

Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A., and Mukhopadhyay, D. A survey on adversarial attacks and defences. *CAAI Transactions on Intelligence Technology*, 6(1):25–45, 2021.

Che, Z., Borji, A., Zhai, G., Ling, S., Li, J., Tian, Y., Guo, G., and Le Callet, P. Adversarial attack against deep saliency models powered by non-redundant priors. *IEEE Transactions on Image Processing*, 30:1973–1988, 2021.

Chen, J., Kao, S.-h., He, H., Zhuo, W., Wen, S., Lee, C.-H., and Chan, S.-H. G. Run, don't walk: Chasing higher flops for faster neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12021–12031, 2023.

Chen, P.-Y., Sharma, Y., Zhang, H., Yi, J., and Hsieh, C.-J. Ead: elastic-net attacks to deep neural networks via adversarial examples. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Chrabaszcz, P., Loshchilov, I., and Hutter, F. A downsampled variant of imagenet as an alternative to the cifar datasets. *arXiv preprint arXiv:1707.08819*, 2017.

Dhillon, G. S., Azizzadenesheli, K., Lipton, Z. C., Bernstein, J., Kossaifi, J., Khanna, A., and Anandkumar, A. Stochastic activation pruning for robust adversarial defense. *arXiv preprint arXiv:1803.01442*, 2018.

Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., and Li, J. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9185–9193, 2018.

Du, C., Huo, C., Zhang, L., Chen, B., and Yuan, Y. Fast c&w: A fast adversarial attack algorithm to fool sar target recognition with deep convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2021.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Halmosi, L., Mohos, B., and Jelasity, M. Evaluating the adversarial robustness of semantic segmentation: Trying harder pays off. In *European Conference on Computer Vision*, pp. 1–18. Springer, 2024.

Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., Tang, Y., Xiao, A., Xu, C., Xu, Y., et al. A survey on vision transformer. *IEEE transactions on pattern analysis and machine intelligence*, 45(1):87–110, 2022.

Han, S., Mao, H., and Dally, W. J. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

Harry24k. adversarial-attacks-pytorch. `https://github.com/Harry24k/adversarial-attacks-pytorch`, 2024. Accessed: Jan 2024.

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

He, Y. and Xiao, L. Structured pruning for deep convolutional neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 2023.

Huang, Q., Katsman, I., He, H., Gu, Z., Belongie, S., and Lim, S.-N. Enhancing adversarial example transferability with an intermediate level attack. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4733–4742, 2019.

Hussain, H., Tamizharasan, P., and Rahul, C. Design possibilities and challenges of dnn models: a review on the perspective of end devices. *Artificial Intelligence Review*, pp. 1–59, 2022.

Jordao, A. and Pedrini, H. On the effect of pruning on adversarial robustness. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1–11, 2021.

Khan, A., Rauf, Z., Sohail, A., Khan, A. R., Asif, H., Asif, A., and Farooq, U. A survey of the vision transformers and their cnn-transformer based variants. *Artificial Intelligence Review*, 56(Suppl 3):2917–2970, 2023.

Kim, H. Torchattacks: A pytorch repository for adversarial attacks. *arXiv preprint arXiv:2010.01950*, 2020.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Li, C., Wang, H., Yao, W., and Jiang, T. Adversarial attacks in computer vision: a survey. *Journal of Membrane Computing*, pp. 1–18, 2024.

Li, H., Li, G., and Yu, Y. Rosa: Robust salient object detection against adversarial attacks. *IEEE transactions on cybernetics*, 50(11):4835–4847, 2019.

Li, Y., Chen, Y., Dai, X., Chen, D., Liu, M., Yuan, L., Liu, Z., Zhang, L., and Vasconcelos, N. Micronet: Improving image recognition with extremely low flops. In *Proceedings of the IEEE/CVF International conference on computer vision*, pp. 468–477, 2021.

Liang, J., He, R., and Tan, T. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision*, pp. 1–34, 2024.

Lin, S., Ji, R., Yan, C., Zhang, B., Cao, L., Ye, Q., Huang, F., and Doermann, D. Towards optimal structured cnn pruning via generative adversarial learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2790–2799, 2019.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Matyasko, A. and Chau, L.-P. Pdpgd: Primal-dual proximal gradient descent adversarial attack. *arXiv preprint arXiv:2106.01538*, 2021.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

Papernot, N., Faghri, F., Carlini, N., Goodfellow, I., Feinman, R., Kurakin, A., Xie, C., Sharma, Y., Brown, T., Roy, A., et al. Technical report on the cleverhans v2. 1.0 adversarial examples library. *arXiv preprint arXiv:1610.00768*, 2016a.

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

Papernot, N., McDaniel, P., Wu, X., Jha, S., and Swami, A. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE symposium on security and privacy (SP)*, pp. 582–597. IEEE, 2016c.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Poursaeed, O., Katsman, I., Gao, B., and Belongie, S. Generative adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4422–4431, 2018.

Qin, Y., Xiong, Y., Yi, J., and Hsieh, C.-J. Training meta-surrogate model for transferable adversarial attack. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 9516–9524, 2023.

Restuccia, G. Semantic extraction attacks. `https://github.com/gabrielication/semantic_extraction_attacks`, 2025. Accessed: 2025-05-16.

Ribani, R. and Marengoni, M. A survey of transfer learning for convolutional neural networks. In *2019 32nd SIBGRAPI conference on graphics, patterns and images tutorials (SIBGRAPI-T)*, pp. 47–57. IEEE, 2019.

Santos, J. F., Huff, A., Campos, D., Cardoso, K. V., Both, C. B., and DaSilva, L. A. Managing o-ran networks: xapp development from zero to hero. *IEEE Communications Surveys & Tutorials*, 2025.

Sayyed, S., Ashdown, J., and Restuccia, F. Faster and accurate neural networks with semantic inference. *arXiv preprint arXiv:2310.01259*, 2023.

Schwinn, L., Raab, R., Nguyen, A., Zanca, D., and Eskofier, B. Exploring misclassifications of robust neural networks to enhance adversarial attacks. *Applied Intelligence*, pp. 1–17, 2023.

Sehwag, V., Wang, S., Mittal, P., and Jana, S. Hydra: Pruning adversarially robust neural networks. *Advances in Neural Information Processing Systems*, 33:19655–19666, 2020.

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

Talaei Khoei, T., Ould Slimane, H., and Kaabouch, N. Deep learning: Systematic review, models, challenges, and research directions. *Neural Computing and Applications*, 35(31):23103–23124, 2023.

Tan, C. M. J. and Motani, M. Dropnet: Reducing neural network complexity via iterative pruning. In *International conference on machine learning*, pp. 9356–9366. PMLR, 2020.

Todi, A., Narula, N., Sharma, M., and Gupta, U. Convnext: A contemporary architecture for convolutional neural networks for image classification. In *2023 3rd International Conference on Innovative Sustainable Computational Technologies (CISCT)*, pp. 1–6. IEEE, 2023.

Vasu, P. K. A., Gabriel, J., Zhu, J., Tuzel, O., and Ranjan, A. Mobileone: An improved one millisecond mobile backbone. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7907–7917, 2023.

Vemparala, M.-R., Fasfous, N., Frickenstein, A., Sarkar, S., Zhao, Q., Kuhn, S., Frickenstein, L., Singh, A., Unger, C., Nagaraja, N.-S., et al. Adversarial robust model compression using in-train pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 66–75, 2021.

Wang, Y., Yao, Q., Kwok, J. T., and Ni, L. M. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)*, 53(3):1–34, 2020.

Wang, Y., Sun, T., Li, S., Yuan, X., Ni, W., Hossain, E., and Poor, H. V. Adversarial attacks and defenses in machine learning-powered networks: A contemporary survey. *arXiv preprint arXiv:2303.06302*, 2023.

Wang, Z., Yang, Y., Li, J., and Zhu, X. Universal adversarial perturbations generative network. *World Wide Web*, 25 (4):1725–1746, 2022.

Wu, D. and Wang, Y. Adversarial neuron pruning purifies backdoored deep models. *Advances in Neural Information Processing Systems*, 34:16913–16925, 2021.

Wu, D., Wang, Y., Xia, S.-T., Bailey, J., and Ma, X. Skip connections matter: On the transferability of adversarial examples generated with resnets. *arXiv preprint arXiv:2002.05990*, 2020.

Yao, Z., Gholami, A., Xu, P., Keutzer, K., and Mahoney, M. W. Trust region based adversarial attack on neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11350–11359, 2019.

Ye, S., Xu, K., Liu, S., Cheng, H., Lambrechts, J.-H., Zhang, H., Zhou, A., Ma, K., Wang, Y., and Lin, X. Adversarial robustness vs. model compression, or both? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 111–120, 2019.

Zhang, H., Avrithis, Y., Furon, T., and Amsaleg, L. Walking on the edge: Fast, low-distortion adversarial examples. *IEEE Transactions on Information Forensics and Security*, 16:701–713, 2020.

Zhang, Z. and Wu, T. Learning ordered top-k adversarial attacks via adversarial distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 776–777, 2020.