arXiv:2509.20148v1 [cs.CV] 24 Sep 2025

Smaller is Better: Enhancing Transparency in Vehicle AI Systems via Pruning

Sanish Suwal
Rochester Institute of Technology

Shaurya Garg
Independent Researcher

Dipkamal Bhusal
Rochester Institute of Technology

Michael Clifford
Toyota InfoTech Labs

Nidhi Rastogi
Rochester Institute of Technology

Abstract

Connected and autonomous vehicles continue to heavily rely on AI systems, where transparency and security are critical for trust and operational safety. Post-hoc explanations provide transparency to these black-box like AI models but the quality and reliability of these explanations is often questioned due to inconsistencies and lack of faithfulness in representing model decisions. This paper systematically examines the impact of three widely used training approaches, namely natural training, adversarial training, and pruning, affect the quality of post-hoc explanations for traffic sign classifiers. Through extensive empirical evaluation, we demonstrate that pruning *significantly* enhances the comprehensibility and faithfulness of explanations (using saliency maps). Our findings reveal that pruning not only improves model efficiency but also enforces sparsity in learned representation, leading to more interpretable and reliable decisions. Additionally, these insights suggest that pruning is a promising strategy for developing transparent deep learning models, especially in resource-constrained vehicular AI systems.

1 Introduction

The increasing reliance on artificial intelligence (AI) in vehicles, such as autonomous cars, drones, and other connected systems, has transformed transportation by enabling capabilities like real-time navigation, vehicle-to-everything (V2X) communication, and intelligent decision-making [22]. Despite the fact that AI in vehicles promises enhanced autonomy and safety, the inherent opacity or "black-box" nature of these models challenges transparency and trust, creating potential security and compliance risks in safety-critical settings. [27].

In response to this challenge, research in explainable AI (XAI) has intensified, especially in developing *post-hoc explanation* methods to foster trust, ensure compliance with safety standards, and facilitate debugging in adverse scenarios [4]. Post-hoc explanation methods such as LIME [27], SHAP [24], Vanilla Gradient [32], which interpret trained models by attributing importance to each input features and identifying the most influential ones. However, these methods are not without limitations. Perturbation-based methods like LIME and SHAP are shown to be unreliable in high-dimensional inputs like images [33], and several explanation methods produce unfaithful explanations [2]. Similarly, gradient-based methods are sensitive to hyperparameter choices and frequently fail standard interpretability tests, leading to misleading explanations [2, 3].

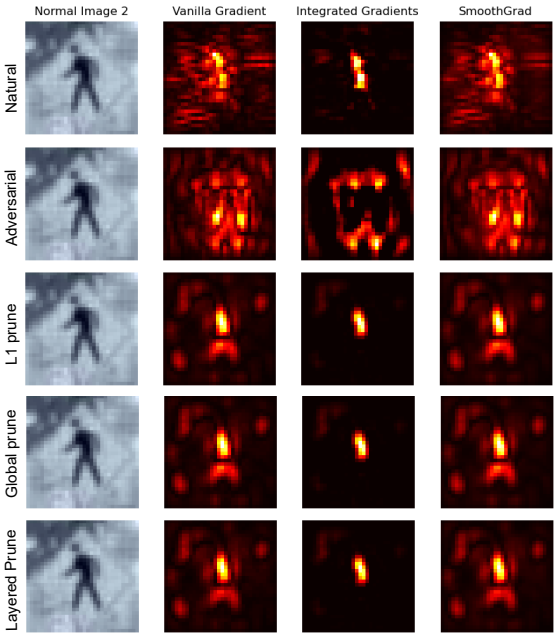

Figure 1: A figure showing heatmaps on an American traffic sign for naturally trained, adversarially trained and different pruned models for three explanation methods: Vanilla Gradient [32], Integrated Gradient [38] and SmoothGrad [34]

Given these limitations, it is critical that post-hoc explanations are reliable, trustworthy, and comprehensible. In addition, they should ensure their practical utility by satisfying some key quantitative properties: sparsity [7] and faithful-

ness [28]. *Sparsity* measures if the explanations focus on the most relevant features by discarding irrelevant ones, thus improving comprehensibility. *Faithfulness* property requires that the post-hoc explanations accurately reflect the model's actual decision-making process, thus explaining what the model actually did in making the prediction. While methods like Guided Backpropagation (GBP) [35], Integrated Gradient (IG) [38], SmoothGrad [34], NoiseGrad [6] have been proposed to improve the quality of explanations, they remain sensitive to change in their hyperparameters [3], and fail several sanity tests [2, 21].

In this work, we take a complementary approach for improving the quality of post-hoc explanations. Recent works have shown that adversarial training [14] can improve the comprehensibility of saliency maps by encouraging models to rely on more robust and semantically meaningful features in high-dimensional images [10]. However, adversarial training is computationally expensive, requiring generation of adversarial samples and additional training overhead often leading to performance degradation on benign inputs. This raises a question: **Can we achieve similar improvements in explanation quality without the drawbacks of adversarial training?**

Pruning [16], traditionally used for model compression, offers an alternative. By removing less significant parameters, pruning not only reduces model complexity but also enforces sparsity in learned representations. We hypothesize that this sparsity can significantly improve the interpretability and reliability of saliency-based explanations by highlighting critical decision-making features. Unlike adversarial training, pruning is lightweight, can be applied both pre- *and* post-training, and is more suitable for deployment in resource-constrained environments like autonomous vehicles.

To address this gap, we systematically analyze the impact of three widely used training approaches: *natural, adversarial and pruning*. We measure saliency map quality for traffic sign recognition models trained on multiple datasets, namely the American traffic sign dataset, LISA [23] and the German traffic sign dataset, GTSRB [36]. Our extensive empirical evaluation reveals that pruning enhances the comprehensibility of saliency maps, which improves their faithfulness, thereby reflecting the underlying decision-making of the model. Figure 1 shows saliency maps on a traffic sign for different models using three explanation methods, where we can clearly observe that pruning improves the interpretability of saliency maps by focusing on critical features of the input image.

Key contributions: While pruning is primarily used for model compression and efficiency, its role in enhancing interpretability remains unclear. This work aims to inspect how pruning impacts the sparsity and faithfulness of saliency maps by conducting extensive experiments on models trained on the LISA and GTSRB traffic sign datasets.

Our main contributions are as follows:

1. **Comprehensive evaluation of explanations:** In Section 4, we systematically compare natural training, adversarial training, and different pruning techniques to assess their impact on saliency map interpretability of vehicular AI model. We perform both quantitative and qualitative evaluations that provide an in-depth assessment of explanation quality for different training strategies.

2. **New insights on pruning for interpretability:** Our results in Section 5 demonstrate that pruning enhances faithfulness and sparsity of saliency maps, making them more human-comprehensible and reliable. While adversarial training improves robustness of models, it often leads to noisier explanations, particularly in low-resolution datasets like LISA and GTSRB. This suggests that pruning, in addition to model compression, can be a more effective approach to improving interpretability without sacrificing model performance.

3. **Guidelines for model interpretability enhancement:** By demonstrating that pruning enhances interpretability, we provide an alternative pathway to model transparency. Pruning is a technique that is primarily used for model compression, however, its advantage in model transparency as demonstrated in our work, makes pruning particularly valuable in safety-critical scenarios, where human-comprehensible explanations are necessary for regulatory compliance and trustworthiness.

The rest of the paper is structured as follows: Section 2 describes the background on explainable AI, explanation methods and their evaluation metrics, and different model training strategies. Section 4 discusses the methodology of the experiment that includes datasets, model training, and metrics. Section 5 presents the qualitative and quantitative evaluation of saliency maps, and discusses key findings, and implications. Section 7 concludes the paper with recommendations for model selection and interpretability enhancements.

2 Background and Related Work

2.1 Explainable AI- Methods and Challenges

Interpretability in deep learning (DL) models focuses on two primary research directions: (1) designing intrinsically interpretable models, and (2) developing post-hoc explanation methods. While the former focuses on building models whose structure and decisions are inherently explainable, the latter involves techniques that provide insights into the decisions of complex, opaque models by analyzing their predictions in relation to input features [4]. Building inherently interpretable models with high performance for complex datasets like images, videos and texts is a challenging task, hence, post-hoc methods have gained significant traction.

Post-hoc explanation methods can be categorized based on several factors, including the granularity of explanations (local vs. global), the type of models they support (model-agnostic vs. model-specific), and the nature of the explanations they provide (feature attribution, rule-based, or counterfactual explanations). **Feature attribution methods** assign relevance scores to individual features, indicating their importance in the model's prediction for a given instance. Such scores are visualized as saliency maps in image classifiers. Formally, given a black-box model $F(.)$ and a test instance $\mathbf{x} = \{x_1, x_2, \ldots, x_N\}$, where N represents the number of input features, a feature attribution method returns a vector $\phi(\mathbf{x})$ with the same dimension as $\mathbf{x}$ that provides the relevance of each feature to the given model prediction $F(\mathbf{x})$. **Perturbation-based feature attribution methods** such as LIME [27] perturb a given instance to generate multiple samples, train a simpler interpretable model (eg. logistic regression) to explain the test instance. **Backpropagation-based attribution methods** (e.g, Integrated Gradient [38]) utilize gradients to propagate the model prediction to the input layer. These methods pass several sanity checks [2], are faster to compute and widely used due to their versatility in applications and ease of interpretation [4]. Hence, we focus on three representative gradient-based post-hoc explanation method: Vanilla Gradient (VG), Integrated Gradient (IG) and SmoothGrad (SG).

Vanilla Gradient (VG) [32]: VG is the one of the earliest gradient-based explanation approach. This technique computes the gradient of the class score with respect to the input features. This gradient value indicates the sensitivity of the output prediction to changes in input features and is used as a feature attribution score, visualized as saliency maps (heatmap) for images.

$$VG(\mathbf{x}) = \frac{\partial F(\mathbf{x})}{\partial x_i} \tag{1}$$

Here, $F(\mathbf{x})$ is the model's output function, and x_i represents an input feature. While simple and computationally efficient, Vanilla Gradient method can produce noisy explanations [34], especially for complex neural networks, and suffers from gradient saturation [30].

Integrated Gradients (IG) [38]: IG builds upon Vanilla Gradient to address its limitations, particularly the issue of noisy attributions due to gradient saturation. IG introduces the concept of a baseline input (e.g., a black image for image-based models) and computes gradients along a straight path from this baseline to the actual input. By integrating these gradients, IG determines the contribution of each input feature to the output while satisfying two key axioms:

1. *Sensitivity:* If a single input feature differs between the baseline and the input, and this difference affects the model's prediction, the attribution for that feature must be non-zero.
2. *Implementation Invariance:* Two functionally equivalent models (even with different implementations) should yield identical attributions for the same input.

These axioms ensure that IG provides robust and reliable attributions, addressing deficiencies in earlier methods. The IG formulation is given as:

$$IG_i(x) = (x_i - x_i'). \int_{\alpha=0}^{1} \frac{\partial F(\mathbf{x}' + \alpha(\mathbf{x} - \mathbf{x}'))}{\partial x_i} \partial\alpha \tag{2}$$

Here, $\mathbf{x}'$ is the baseline, $\mathbf{x}$ is the input, and α controls the transition between them, varying from 0 to 1. This method has become one of the most widely used gradient-based approaches for feature attribution. The selection of an appropriate baseline depends on the context and dataset [37], and can significantly influence the quality of the explanations.

SmoothGrad (SG) [34]: To further address the noise in gradient-based saliency maps, Smilkov et al. introduced SmoothGrad. This method improves explanation stability by creating n perturbed versions of the input through the addition of Gaussian noise ($N(0, \sigma^2)$). Gradients are computed for each perturbed sample, and their average is taken to produce the final attribution map. This averaging reduces noise, highlights meaningful patterns, and generates smoother, more interpretable explanations. The formulation for SmoothGrad is as follows:

$$SG_i(x) = \frac{1}{N} \sum_{k=1}^{N} \frac{\partial F(\mathbf{x} + N(0, \sigma^2))}{\partial x_i} \tag{3}$$

Here, $N(0, \sigma^2)$ represents Gaussian noise. SmoothGrad often outperforms basic gradient-based methods by suppressing spurious artifacts in explanations.

2.2 Evaluating Explanation Quality

Qualitative evaluation of saliency maps are subjected to human-bias and makes it difficult to judge whether an explanation is correct. Quantitative evaluation of saliency maps, however, utilize formal definitions and properties of explanation quality, and does not require human validation. Below, we explain the evaluation metrics of saliency maps used in our evaluation:

2.2.1 Measuring Explanation Sparsity

Sparsity measures how focused an attribution vector $\phi(\mathbf{x})$ is by evaluating the distribution of importance across input features. We use the Gini index to quantify sparsity [7]. For an attribution vector $\phi(\mathbf{x}) \in \mathbb{R}^d$, the elements are first sorted in non-decreasing order, and the Gini index is computed as follows:

$$G(\phi(\mathbf{x})) = 1 - 2\sum_{k=1}^{d} \frac{\phi(\mathbf{x})_{(k)}}{||\phi(\mathbf{x})||_1} \frac{d-k+0.5}{d} \tag{4}$$

Here, $||\phi(\mathbf{x})||_1$ is the L_1-norm of $\phi(\mathbf{x})$, and $\phi(\mathbf{x})_{(k)}$ denotes the k-th smallest element in the sorted vector. The Gini index ranges from 0 to 1. A value of 1 indicates perfect sparsity, where only one element in $\phi(\mathbf{x})$ is non-zero. A value of 0 indicates uniform distribution across all features.

Sparsity helps evaluate how concentrated the attribution scores are, with higher sparsity leading to more comprehensible and human-friendly explanations.

2.2.2 Measuring Explanation Faithfulness

Faithfulness measures the correctness of an explanation method in capturing relevant features for a given test sample. This is the most crucial quantitative metric as we want explanations to truly represent the model we want to explain. We measure faithfulness using ROAD metric [28]. ROAD (Remove and Debias) evaluates the accuracy of a model as the most important features are iteratively removed. Using the MoRF (Most Relevant First) removal strategy, features are ordered by decreasing importance based on an attribution method, and the top k features are removed in each iteration. The accuracy is tracked at each step.

Given a model F, input sample $\mathbf{x}$, and an attribution method that assigns importance scores to features, the removal process is performed iteratively. The removal uses noisy linear imputation to prevent out-of-distribution samples. For our experiments, we set $k = 5$.

A sharper accuracy drop as features are removed indicates a better explanation, as the most relevant features have a greater impact on model predictions. ROAD is preferred over other faithfulness metrics like Insertion/Deletion [26] or ROAR [19], as it avoids distribution shifts caused by perturbations and does not require expensive model retraining.

2.3 Model training strategies

2.3.1 Natural Training

Natural training refers to the conventional optimization process where a model is trained by minimizing a loss function L, typically using gradient descent or its variants [13]. For a model F, input $\mathbf{x}$, true label y, and parameters θ, the objective is to find θ that minimizes the empirical risk:

$$\theta^* = \arg\min_{\theta} \mathbb{E}_{(x,y)\sim D}\big[L(F(\mathbf{x};\theta),y)\big] \tag{5}$$

Here, D represents the data distribution. In this framework, the model learns to generalize by optimizing performance on the training data without any explicit mechanisms to handle robustness or interpretability. While effective in many scenarios, natural training often results in over-parameterized networks susceptible to adversarial attacks [14] and challenges in post-hoc explanation quality due to noise and saturation in gradients [34].

2.3.2 Adversarial Training

Adversarial training enhances a model's robustness against adversarial attacks by training it on perturbed inputs [14]. Given an input $\mathbf{x}$ and its true label y, an adversarial example $\mathbf{x}'$ is crafted to maximize the model's loss:

$$\mathbf{x}' = \arg\max_{\mathbf{x}'\in B_\varepsilon(\mathbf{x})} L(F(\mathbf{x}';\theta),y) \tag{6}$$

Here, $B_\varepsilon(\mathbf{x})$ is an ε-ball around x, representing the set of valid perturbations constrained by $||\mathbf{x}'-\mathbf{x}|| \le \varepsilon$. Adversarial training involves minimizing the worst-case loss:

$$\theta^* = \arg\min_{\theta} \mathbb{E}_{(\mathbf{x},y)\sim D}\Big[\max_{\mathbf{x}'\in B_\varepsilon(x)} L(F(\mathbf{x}';\theta),y)\Big] \tag{7}$$

This approach improves the model's resilience to malicious inputs, such as spoofed sensor data in vehicular systems. Recent studies have shown that post-hoc explanations of adversarially trained models are clearer and sparser than naturally trained models [7, 10, 43]. However, its impact on explanation quality of vehicular system has not been thoroughly explored.

2.3.3 Neural Network Pruning

Large neural networks consume more memory space, require more time for computation, and present challenge for deployment on devices where computational resources is limited such as autonomous driving [9, 15, 16]. Pruning is an effective technique of compressing neural network to save memory space and inference time computation. It reduces the number of parameters in a model to decrease computational requirements and enhance efficiency in constrained environments. By removing redundant weights, neurons, or filters, pruning can also improve inference speed, energy efficiency, and, in some cases, model accuracy and robustness [8]. We explore following three types of neural network pruning:

1. Unstructured pruning: Given a neural network with weights $W = \{w_0, w_1, ..., w_K\}$, a dataset $D = \{(x_i, y_i)\}_{i=1}^N$ consisting of input-output pairs (x_i, y_i), and a target number of non-zero weights k (where $k < K$), unstructured pruning can be formulated as the following constrained optimization problem [8]:

$$\min_W L(W;D) = \min_W \frac{1}{N}\sum_{i=1}^{N} \ell(W;(x_i,y_i)), \quad \text{s.t.} \quad \|W\|_0 \le k. \tag{8}$$

In practice, for small to medium-sized models, unstructured pruning does not directly set weights to zero. Instead, it applies a binary mask M that determines which weights are active, setting the masked-out weights to zero [40]. Typically, after pruning, the network undergoes retraining, either

with fine-tuning or training from scratch, while keeping the mask M fixed. The pruned weights (those masked out) are not updated during this process.

2. Structured pruning: Structured pruning removes entire structural components of a neural network, such as channels, filters, neurons, or transformer attention heads, to achieve a targeted pruning ratio while minimizing performance degradation and maximizing speed improvements [18]. Structured pruning are hardware-friendly for deployment since they maximize speed improvement.

Formally, given a neural network with structural components $S = \{s_1, s_2, ..., s_L\}$, where each s_i represents a set of channels, filters, neurons, or attention heads in layer i, structured pruning searches for a pruned set $S' = \{s'_1, s'_2, ..., s'_L\}$ such that:

$$s'_i \subseteq s_i, \quad i \in \{1, ..., L\} \tag{9}$$

3. Global pruning: Global pruning removes the least important parameters across the entire neural network rather than restricting pruning to individual layers or structures. This approach enables a more efficient allocation of remaining parameters by preserving the most critical weights regardless of their layer.

Formally, given a neural network with weights $W = \{w_1, w_2, ..., w_K\}$ and a target number of nonzero weights k (where $k < K$), global pruning searches for a pruned weight set $W' \subset W$ that satisfies:

$$\|W'\|_0 \leq k, \quad \text{where} \quad W' = \{w \in W \mid \text{top-}k(|W|)\} \tag{10}$$

Unlike structured pruning, which removes entire channels, filters, or neurons, global pruning removes individual weights across all layers based on their importance scores, often determined by magnitude, gradients, or other criteria [5, 16].

Pre-train and post-train pruning: Neural network pruning techniques we just discussed can be done either before training (pre-train) or after training (post-train). Pre-train pruning prune neural networks at the very beginning before any training occurs to eliminate the computational cost of pre-training [8]. Let $F(x; W_0 \odot M)$ represent a neural network, where W_0 denotes the initial, randomly sampled weights from a given initialization distribution, and M is a binary mask that determines which weights remain after pruning. Once the pruning is applied, the resulting sparse network $F(x; W_0 \odot M)$ is trained directly from scratch. After t training epochs, the network converges to $F(x; W_t \odot M)$, where W_t represents the modified weights.

Post-train pruning is the most widely used pruning approach, where we first train a randomly initialized dense neural network $F(x; W_0)$ until it converges to $F(x; W_t)$. Then, we remove weights, filters, or neurons that contribute the least to the model's performance. The resulting pruned model is represented as $F(x; W'_t \odot M')$, where W'_t and M' denote the remaining weights and the corresponding pruning mask. This pruning step can be performed once (one-shot pruning), where all unimportant weights are pruned in a single step, multiple times (iterative pruning), where pruning is done gradually over several steps, and retraining (either fine-tune the pruned network $F(x; W'_t \odot M')$ to restore performance, or train the remaining weights $F(x; W_0 \odot M'')$ from scratch, where M'' is the final sparsity pattern after pruning.

The lottery ticket hypothesis [11] suggests that pruned models can achieve higher accuracy than their unpruned counterparts when retrained with their initial weights. Pruning has proven particularly beneficial for real-time applications like autonomous vehicles, where inference speed and energy efficiency are critical.

2.4 Related Work

Recent works have shown some relationship between pruning and post-hoc explanations. For example, Weber et al. [41] evaluate the impact of pruning on explainability. Using Grad-CAM [29] in VGG-16 [31], the authors explore the effects of varying compression rates on saliency maps. Their findings suggest that moderate pruning levels enhance explainability by reducing noise in saliency maps, while excessive pruning negatively affects attribution quality. However, their evaluation lacks a rigorous quantitative comparison of heatmaps and does not consider scenarios like pruning without fine-tuning or pre-train pruning. This leaves significant gaps in understanding how different pruning strategies influence explanation robustness.

Tan et a. [39] investigate the effect of pruning on the robustness of explanations. The study prunes neurons identified as least important post-training, without fine-tuning, ensuring that model predictions remain unchanged. The results demonstrate that removing these neurons can lead to a collapse in explanations generated by XAI methods, even when predictions are unaffected. Khakzar et al. [20] introduce an innovative approach where neural networks are pruned dynamically for individual inputs, retaining only neurons that significantly contribute to the prediction. While this approach highlights the potential of input-specific pruning, it may overlook the need for local importance when explaining individual samples. Additionally, the computational cost of input-specific pruning limits its scalability for large datasets and real-time applications. Abbasi et al. [1] focus on simple pruning of filters in convolutional neural networks (CNNs) to improve interpretability. By selectively removing redundant filters, the approach simplifies network architecture, reducing noise in saliency maps and emphasizing critical features.

While there is existing research on post-hoc explanation methods and training strategies independently, their combined impact on interpretability in vehicular AI remains underexplored. This study addresses this gap by systematically evaluating how training strategies (natural, adversarial, and prun-

ing) affect explanation quality across different models and datasets relevant to vehicular applications. These insights contribute to designing secure and transparent AI systems for vehicles, balancing interpretability with performance.

3 Motivation

In this section, we provide the motivation behind using pruning as a technique for improving post-hoc explanations by discussing its impact on faithfulness and comprehensibility of explanations.

A critical requirement for faithful explanations is that saliency maps must reliably reflect the model's underlying decision-making process. In input-gradient-based methods such as Vanilla Gradient [32], Integrated Gradients [38], and SmoothGrad [34], explanations are computed as the gradient of the model output with respect to input features:

$$\mathcal{E}(\mathbf{x}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \tag{11}$$

where, the explanation $E(x)$ is computed as the gradient of the model's output with respect to the input features and $f(x)$ represents the model's prediction for input x. The gradient highlights how sensitive the model's prediction is to changes in each input feature.

These explanations are then visualized as saliency maps $\mathcal{E}(\mathbf{x})$ that highlight input features that contribute most to the model's decision. The faithfulness of these explanations hence depend on how well the gradient signal captures meaningful model behavior.

For naturally trained deep models, these gradients are often highly sensitive to small input perturbations, leading to noisy and unfaithful explanations. This is quantified by the gradient norm, which measures how much the model's output fluctuates with respect to input perturbations:

$$\|\mathcal{E}(\mathbf{x})\|_2 = \left\| \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right\|_2 \tag{12}$$

If this norm is large, explanations are unstable and influenced by high-frequency artifacts rather than meaningful features. Prior work suggests that adversarial training reduces the gradient norm [12], thereby producing explanations that focus on more stable, robust features [10].

Pruning inherently reduces model complexity by removing unnecessary weights, which stabilizes gradients and mitigates noise, thus improving the clarity and faithfulness of explanations. This results in a smoother, lower-complexity function $f(\mathbf{x})$, which in turns leads to a more stable gradient distribution, ensuring that the saliency maps better capture true decision-making features. Mathematically, pruning results in:

$$\|\mathcal{E}_{\text{pruned}}(\mathbf{x})\|_2 < \|\mathcal{E}_{\text{natural}}(\mathbf{x})\|_2$$

Similarly, for an explanation to be useful, it must be sparse, focusing only on the most critical input features while ignoring irrelevant ones. This affects the comprehensibility of saliency maps. Pruning encourages weight sparsity in the network, which translates to fewer active neurons during inference. This acts as an implicit regularizer, enforcing an information bottleneck effect. As a result, pruned models tend to rely on a more compact and interpretable set of features, making their saliency maps naturally more comprehensible:

Therefore, we hypothesize that pruning significantly enhances interpretability by yielding saliency maps that are more stable, sparse, and representative of genuine model decisions, compared to natural or adversarial training alone.

4 Experiment

In this section, we describe the dataset, model, explanation methods and evaluation metrics used in our experiment. Our code is available at https://github.com/sanishsuwal7/VehicleSecCopy/.

4.1 Datasets

- **LISA**: The LISA Traffic Sign Dataset [23] is a widely used benchmark for traffic sign detection and recognition in real-world driving environments. Collected from a vehicle-mounted camera in the United States, the dataset consists of 47 US traffic sign types with 7,855 annotations across 6,610 frames, capturing a wide range of real-world variations such as sign sizes (ranging from 6×6 to 167×168 pixels). Each traffic sign is carefully annotated with attributes such as sign type, position, size, occlusion status, and whether the sign is on a side road.
- **GTSRB:** The German Traffic Sign Recognition Benchmark (GTSRB) [36] is another widely used dataset for traffic sign classification with more than 50,000 images of 43 different traffic sign classes, captured under real-world conditions with variations in lighting, occlusion, perspective distortion, and motion blur. Each image is annotated with the corresponding traffic sign label and bounding box, providing a comprehensive benchmark for evaluating deep learning models.

4.2 Model Architectures and Training

4.2.1 Natural training

We use the VGG-16 [31] architecture for LISA and GTSRB dataset. Both models are trained for 100 epochs with a batch size of 128. We optimized the network using Stochastic Gradient Descent (SGD) with a learning rate of 0.01, momentum of 0.9, and weight decay of 5e-4 to prevent overfitting. The loss was computed using a mean-reduction cross-entropy function. In addition, we trained a ResNet-18 model [17] pre-trained

on ImageNet for LISA. The training was conducted for 100 epochs with a batch size of 8, using the Ranger optimizer [42] with a learning rate of 1e-4 and epsilon set to 1e-6 for stable convergence. We applied cross-entropy loss with mean reduction for classification. We discuss the results of ResNet model on LISA dataset in Appendix A.

4.2.2 Adversarial training

To perform adversarial training, we generate adversarial examples that are produced from natural samples $\mathbf{x} \in R^d$ by adding a perturbation vector $\delta \in R^d$. We use the PGD [25] attack to obtain adversarial perturbations. The hyper-parameters of PGD attack in our adversarial training: $\varepsilon = 0.01$, attack step size = $\varepsilon/10$, and number of iterations = 40. We also evaluate the quality of saliency maps when adversarial training strength (ε) is increased to 0.1 for LISA dataset. Other training hyperparameters are kept as explained in Section 4.2.1.

4.2.3 Neural Network Pruning

We perform both **pre-** and **post-train** pruning on the models. While pre-train pruning involves pruning before training and optimizing a model to achieve a weight sparsity, post-train pruning is applied to a fully trained model. We evaluate models with and without fine-tuning following post-train pruning. Following are the types of pruning we evaluate in our work (see Section 2.3.3 for details):

- *L1 unstructured pruning:* It focuses on removing individual weights from the model without regard for the structure of the neural network layers. We prune 20% of weights in all convolutional layers and 10% of weights in the output layers resulting in 19% sparsity.

- *Global pruning:* It selects and removes weights across the entire model, rather than within individual layers or based on specific layer criteria. We use the prune rate of 0.2 for the weights of all the layers achieving 20% sparsity.

- *Layered structured pruning:* It removes entire structures such as neurons, filters, or channels layer by layer. We use a rate of 0.1 for each layer except the last fully connected layer resulting in 10% sparsity. We choose less sparsity limit for layered structured pruning because of high loss in accuracy compared to other pruning methods.

Fine-tuning: To improve model performance after pruning, we apply a fine-tuning strategy to optimize the pruned model while preserving its generalization capability. For both models, we initialize the unfine-tuned model, set up the optimizer similar to Section 4.2.1, and retrain the model for 50 epochs.

Table 1: Model performance of natural training (Nat), adversarial training (Adv), unstructured L1 pruning (L1), global pruning (Global), and layered structured pruning (Layered) on VGG-based model for LISA dataset. Here, Pre-train, and Post-train means pruning before training and after training. FT means fine-tuning.

Method/Model	Nat	Adv	L1	Global	Layered
Pre-train	97.6	97.5	94.5	93.9	91.3
Post-train (FT)	97.6	97.5	95.3	95.4	94.9
Post-train (no FT)	97.6	97.5	94.8	94.7	93.8

Table 2: Model performance of natural training (Nat), adversarial training (Adv), unstructured L1 pruning (L1), global pruning (Global), and layered structured pruning (Layered) on VGG-based model for GTSRB dataset. Here, Pre-train, and Post-train means pruning before training and after training. FT means fine-tuning.

Method/Model	Nat	Adv	L1	Global	Layered
Pre-train	99.9	99.2	98.1	99.5	94.4
Post-train (FT)	99.2	98.6	99.9	99.9	98.1
Post-train (no FT)	99.2	98.6	99.9	99.9	96.9

4.3 Explanation methods and metrics

For each model, we evaluate three widely used gradient-based explanations: Vanilla Gradient [32], Integrated Gradients [38], and SmoothGrad [34]) using sparsity [7], and faithfulness [28]. Sparsity metric is computed as the difference between the given method's score and that of naturally trained model. Higher values are better in sparsity. Faithfulness is computed as the ROAD evaluation plot where sharper drop in accuracy represents faithful explanations. See details in Section 2.1 and Section 2.2.

5 Results and Analysis

Table 1 and 2 presents the classification accuracy of a VGG-based model trained on the LISA and GTSRB traffic sign dataset under natural training, adversarial training and different pruning strategies. As expected, naturally trained model has the highest accuracy compared to all other models. Among pruning strategies, fine-tuning after pruning is crucial for maintaining robust performance. Next, we evaluate the quality of saliency maps using different explanation methods in these models.

5.1 Qualitative analysis

Figure 2 shows that explanations from Vanilla Gradient in naturally trained models are noisy whereas saliency maps using Integrated Gradient are sparser by default. Since SmoothGrad

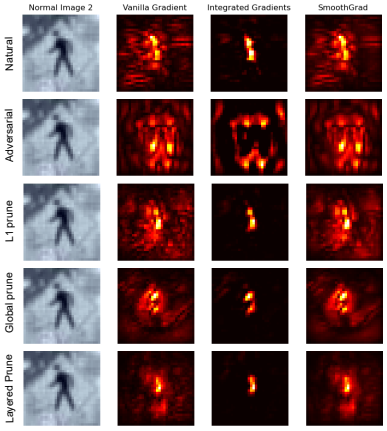

Figure 2: Saliency maps comparison for LISA dataset (VGG) between natural training, adversarial training and pre-train pruning.

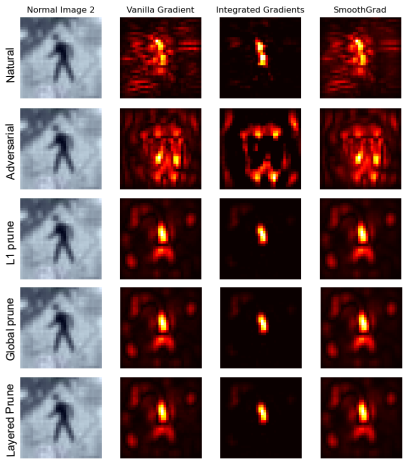

Figure 3: Saliency maps comparison for LISA dataset (VGG) between natural training, adversarial training and test-time pruning with no fine-tuning.

is computed as an average of Vanilla Gradient explanations, the saliency maps are similar to the Vanilla Gradient but with an *averaging* effect. Adversarially trained models produce more noisy saliency maps compared to the naturally trained models. While on larger datasets like ImageNet, adversarial training produces sparser saliency maps [7], this characteristic is not reciprocated with vehicle datasets that consists of low-dimension images. All pruned models, in pre-train pruning, has much sparser saliency maps than adversarially trained models.

In Figure 3, we can observe that saliency maps for pruned models (without fine-tuning) focus on very relevant parts of the pedestrian image. Compared with saliency maps from naturally trained models and adversarially trained models, saliency maps from all three explanation methods, Vanilla

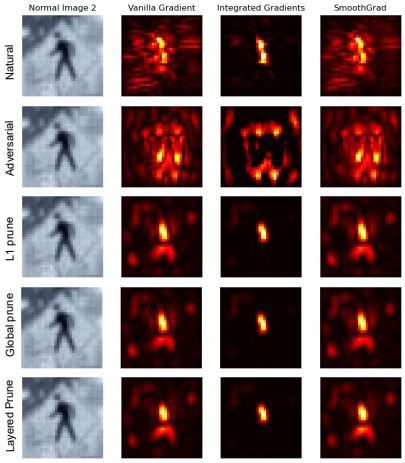

Figure 4: Saliency maps comparison for LISA dataset (VGG) between natural training, adversarial training and test-time pruning with fine-tuning.

Gradient, Integrated Gradient and SmoothGrad, focus on critical human features and are sparser and comprehensible. Similar saliency maps can be observed in Figure 4 where saliency maps with pruned models are sparser and comprehensible.

However, if we increase the adversarial strength $\varepsilon = 0.1$, we get sparser saliency maps with adversarially trained model, as shown in Figure 5. However, pruned models still seem to capture the specific human boundaries in the given image. Quantitatively, the benign accuracy of the model also decreases with increasing adversarial training robustness parameter (in our case, the model performance reduced from 97% to 93% accuracy), and pruned models also have much better faithfulness compared to adversarial trained models. See Section 5.2 for details.

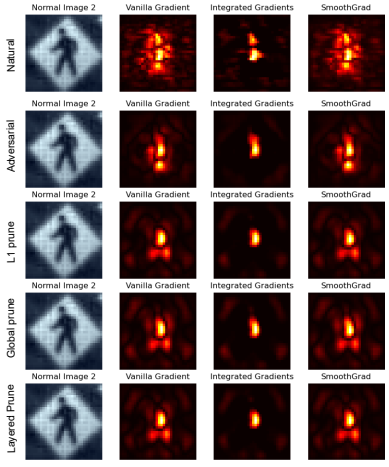

Figure 5: Saliency maps comparison for LISA dataset between natural training, adversarial training ($\varepsilon = 0.1$) and post-train pruning (with fine-tuning) for VGG.

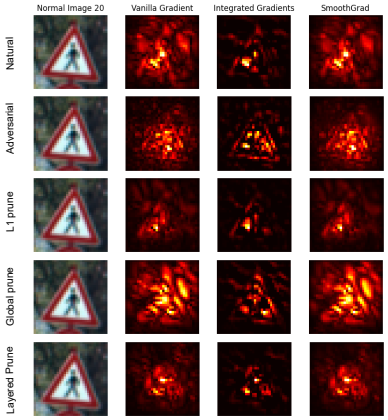

Figure 6: Saliency maps comparison for GTSRB dataset between natural training, adversarial training and pre-train pruning.

Figure 6 shows the saliency maps for GTSRB dataset where we can observe that saliency maps for naturally trained, adversarially trained and global pruned models using Vanilla Gradient are noisy whereas L1 prune and Layered Prune produces much clearer saliency maps. Similar to LISA, saliency maps using Integrated Gradient are sparser by default. However, all pruned models produce more comprehensible saliency maps. In Appendix B, we demonstrate saliency maps comparison for different training strategies and post-train pruning for GTSRB. In Appendix A, we demonstrate saliency maps for different training strategies on ResNet network for LISA dataset, revealing sparse and comprehensible saliency maps for pruned models.

Discussion: The increased clarity of saliency maps in pruned models suggest that pruning encourages feature selectivity by forcing the model to focus on the most critical input features for decision-making. This contrasts with naturally trained models, which, as observed, produce noisier saliency maps in vehicle datasets. One possible explanation is that pruning reduces model redundancy, eliminating unnecessary or redundant parameters that may contribute to spurious attributions in saliency maps. By contrast, natural training only aim to improve accuracy, relying on a broader range of features, thereby producing more noisy explanations.

5.2 Quantitative analysis

While qualitative analysis provides some visual cues of judgement, these are inherently bias to observers. Hence, we perform quantitative evaluation to judge the effectiveness of different training strategies for enhancing explanation quality.

5.2.1 Sparsity of Explanations

Table 3 presents the sparsity scores of Vanilla Gradient (VG), Integrated Gradient (IG), and SmoothGrad (SG) on a VGG-based model trained with various pruning strategies for LISA dataset. The scores are measured using the Gini index and computed as the difference relative to naturally trained models, with higher values indicating better sparsity in attributions.

Table 3: Sparsity evaluation of Vanilla Gradient (VG), Integrated Gradient (IG) and SmoothGrad (SG) on different training strategy of LISA-VGG model. Here, adv means adversarial training and L1, Global and Layered means L1 unstructured pruning, Global pruning and layered structured pruning respectively. Pre-train and post-train means pruning before and after pruning and FT means fine tuning. Higher the better scores.

Method	Model	Adv	L1	Global	Layered
VG	**Pre-train**	-0.03	-0.04	-0.05	-0.21
	Post-train (FT)	-0.03	0.00	0.00	-0.01
	Post-train (no FT)	-0.03	0.00	0.00	0.00
IG	**Pre-train**	-0.05	-0.04	-0.05	-0.25
	Post-train (FT)	-0.05	0.00	-0.01	0.00
	Post-train (no FT)	-0.05	0.00	0.00	0.00
SG	**Pre-train**	-0.04	-0.05	-0.06	-0.21
	Post-train (FT)	-0.04	0.00	0.01	0.00
	Post-train (no FT)	-0.04	0.00	0.00	0.00

Table 4: Sparsity evaluation of Vanilla Gradient (VG), Integrated Gradient (IG) and SmoothGrad (SG) on different training strategy of GTSRB model. Here, adv means adversarial training and L1, Global and Layered means L1 unstructured pruning, Global pruning and layered structured pruning respectively. Pre-train and post-train means pruning before and after pruning and FT means fine tuning. Higher the better scores.

Method	Model	Adv	L1	Global	Layered
VG	**Pre-train**	0.03	0.00	0.00	-0.02
	Post-train (FT)	0.03	0.00	0.00	0.00
	Post-train (no FT)	0.03	0.00	0.00	0.00
IG	**Pre-train**	0.01	0.00	0.00	-0.04
	Post-train (FT)	0.01	0.00	0.00	-0.01
	Post-train (no FT)	0.01	0.00	0.00	0.00
SG	**Pre-train**	0.03	0.00	0.00	-0.02
	Post-train (FT)	0.03	0.00	0.00	-0.01
	Post-train (no FT)	0.03	0.00	0.00	0.00

We can observe from Table 3 that adversarially trained models have consistently negative sparsity scores across all explanation methods, indicating that adversarial training does not improve sparsity and instead makes attributions more distributed compared to naturally trained models. This aligns with qualitative observations from Figures 2, 3, and 4, where adversarial training resulted in noisier saliency maps. However, we can also observe that pruned models do not necessarily have significantly positive sparsity scores; in fact, most pruning strategies have sparsity scores close to zero or slightly negative. This suggests that the overall distribution of feature

Table 5: Sparsity evaluation of Vanilla Gradient (VG), Integrated Gradient (IG) and SmoothGrad (SG) on different training strategy of ResNet model. Here, adv means adversarial training ($\epsilon = 0.1$) and L1, Global and Layered means L1 unstructured pruning, Global pruning and layered structured pruning with fine-tuning. Higher the better scores.

Method	Adv	L1	Global	Layered
VG	-0.01	0.00	0.00	-0.01
IG	-0.03	0.00	-0.01	0.00
SG	-0.01	0.00	0.01	0.00

importance remains similar between naturally trained and pruned models, with no drastic increase in sparsity as measured by the Gini index. However, this quantitative sparsity measurement does not fully capture the qualitative improvements observed in the saliency maps of pruned models. As shown in Figures 3 and 4, the saliency maps from pruned models clearly highlight critical object boundaries, for pedestrian images. This indicates that while pruning does not necessarily lead to a mathematically sparser distribution of attributions, it refines feature selection, leading to qualitatively clearer and more human-comprehensible explanations.

In Table 5, we evaluate the sparsity for saliency maps between adversarially trained models ($\epsilon = 0.1$) and post-train pruning with fine-tuning. We observed in Figure 5 that adversarially trained models produced sparser explanations when adversarial training strength was increased. However, similar to results in Table 3, quantitatively, there is little to no gain in the sparsity scores. Even with increase in comprehensibility of saliency maps from adversarially trained models, these are still less sparse than all pruned models.

Similar results can be observed in Table 4 for GTSRB dataset where adversarially trained models or pruned models do not necessarily have high sparsity. This confirms that these techniques maintain the overall distribution of feature importance but are able to focus on relevant parts of the image.

Discussion: These findings suggest that pruning does not simply increase attribution sparsity, but rather refines the relevance of explanations. It encourages the model to focus on essential object boundaries, leading to more interpretable saliency maps. This supports the hypothesis that pruning acts as an implicit regularizer, reducing redundant connections and forcing the network to make decisions based on a smaller set of critical features.

5.2.2 Faithfulness of Explanations

Faithfulness measures whether the features highlighted by explanations accurately reflect the model's decision-making process. It is a critical metric, as effective explanations should help us understand how the model arrives at its predictions. We evaluate faithfulness using the ROAD (Remove and Debias) metric [28], where features are iteratively removed in order of decreasing importance (Most Relevant First (MoRF)), and the drop in model accuracy is observed. A sharper decline in accuracy indicates higher faithfulness, as it demonstrates that the removed features were indeed critical to the model's predictions.

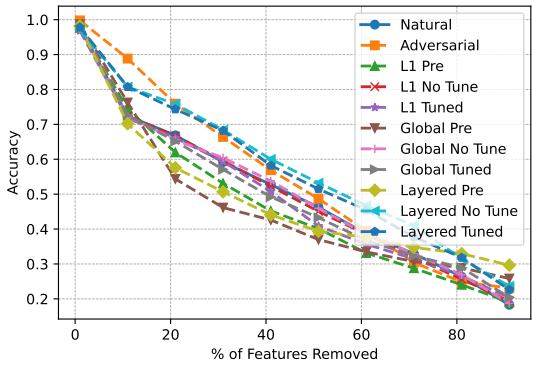

Figure 7: Faithfulness evaluation of Vanilla Gradient using ROAD on VGG for LISA using different strategies.

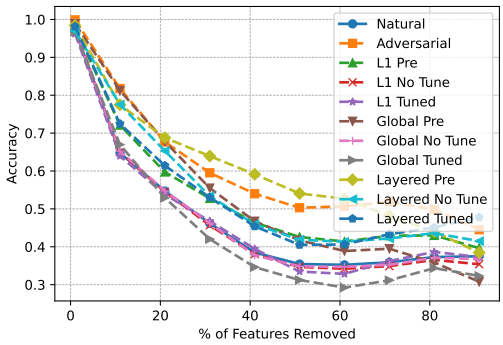

Figure 8: Faithfulness evaluation of Integrated Gradient using ROAD on VGG for LISA using different strategies.

Figure 7 presents the faithfulness evaluation for Vanilla Gradient (VG) explanations across different training strategies. The results indicate that all pruned models (except Layered Pre-Prune) exhibit a steeper accuracy drop compared to naturally trained and adversarially trained models, signifying that pruning improves faithfulness. All pruning methods, in particular, leads to the most faithful explanations, as evidenced by a steeper accuracy decline within the 0–40% feature removal range, meaning that the most relevant features identified by the explanations have a strong impact on model predictions. In contrast, naturally trained and adversarially trained models show a slower decay in accuracy, suggesting that their explanations do not strongly align with the model's actual decision process, making them less faithful. These findings align with our earlier sparsity analyses, where we observed that naturally and adversarially trained models produced noisier explanations, making explanations less focused,

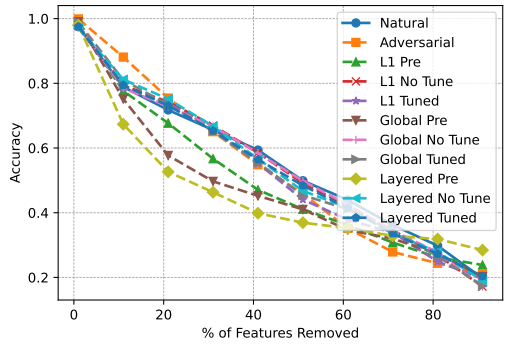

Figure 9: Faithfulness evaluation of SmoothGrad using ROAD on VGG for LISA using different strategies.

contributing to their weaker faithfulness in the ROAD evaluation. Similar trends are observed in Figure 8, which shows the faithfulness evaluation for Integrated Gradient (IG) explanations. The results confirm that pruning enhances explanation faithfulness, particularly for global fine-tuned pruning, which exhibits the steepest accuracy drop among all methods.

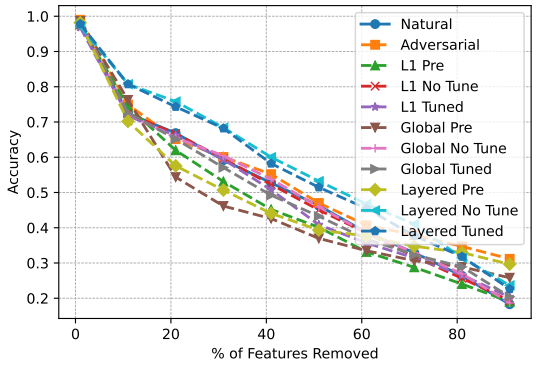

Figure 10: Faithfulness evaluation of Vanilla Gradient for LISA (adversarial training strength $\varepsilon = 0.1$)

Figure 9 shows the faithfulness evaluation for SmoothGrad (SG) explanations, further validating that pruning enhances the faithfulness of saliency maps. The ROAD curves for Layered Pre-Prune, Global Pre-Prune, and L1 Pre-Prune (pre-train pruning methods) exhibit the steepest accuracy declines, demonstrating that pre-train pruning significantly improves the faithfulness of explanations. In Appendix A, we extend our faithfulness analysis to ResNet models, where we observe that the naturally trained ResNet model has an almost flat ROAD curve, signifying that the explanations generated from the naturally trained model are not faithful to the model's actual decision process. This further supports our finding that pruning forces the model to rely on a subset of essential features, thereby improving explanation faithfulness and interpretability.

As discussed before, increasing the adversarial training

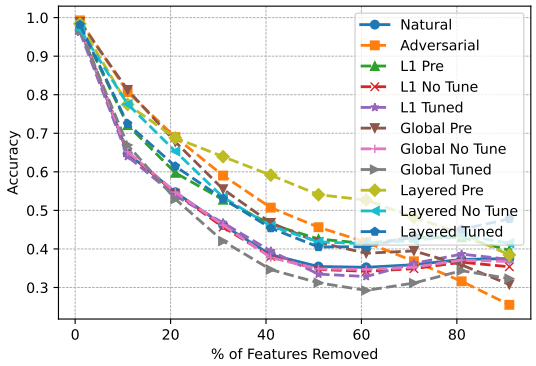

Figure 11: Faithfulness evaluation of Integrated Gradient LISA (adversarial training strength $\varepsilon = 0.1$)

strength led to less noisy saliency maps. However, we demonstrated that this does not lead to any quantitative improvement in sparsity scores (see Table 5). Figures 10, 11 and 12 demonstrate that pruning techniques, especially pre-train pruning, consistently emerge as the most effective strategies for generating faithful explanations, evident by the steepest drop in accuracy compared to adversarialy and naturally trained models. While there is an improvement over faithfulness explanations with adversarially trained model, pruning still generates the most faithful explanations.

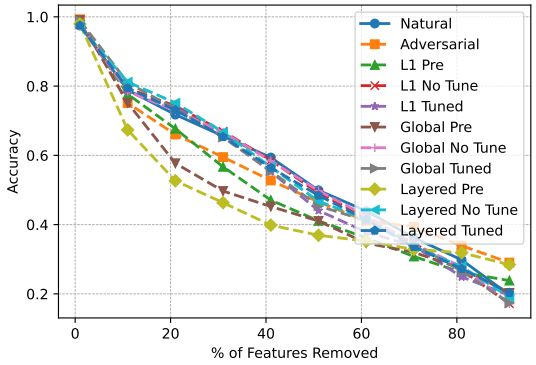

Figure 12: Faithfulness evaluation of Smooth Gradient LISA (adversarial training strength $\varepsilon = 0.1$)

These faithfulness results are consistent in GTSRB model. As shown in Figure 13, in Vanilla Gradient, the adversarial models are the least faithful while the faithful measures in naturally trained models overlap with the pruned models. This is consistent with Integrated Gradient in Figure 14, and SmoothGrad in Figure 15.

Discussion: Pruning techniques consistently emerge as the most effective strategies for generating faithful explanations, indicating their ability of using the most important features in an input. This demonstrates that pruning not only reduces model complexity but also enhances the interpretability of

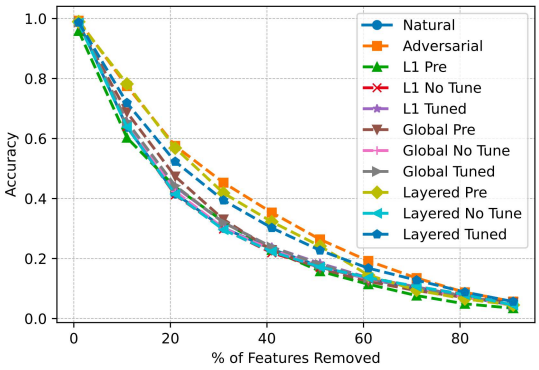

Figure 13: Faithfulness evaluation of Vanilla Gradient using ROAD for GTSRB using different strategies.

AI systems. The faithfulness evaluation also underscores that explanations from naturally trained model are consistently less faithful meaning we cannot rely on post-hoc explanations of such models to understand a model prediction, and require training-strategy intervention for obtaining reliable explanations.

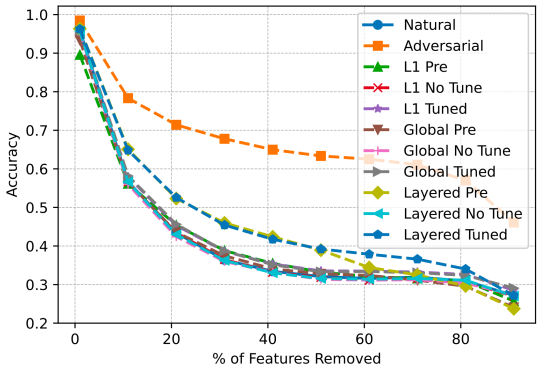

Figure 14: Faithfulness evaluation of Integrated Gradient using ROAD for GTSRB using different strategies.

6 Limitations

We conducted our experiments on two popular traffic sign datasets: LISA and GTSRB and demonstrated how pruning a model can lead to comprehensible and faithful explanations improving model efficiency which is suitable for vehicular systems. However, while we explored different pruning techniques, the choice of the optimal pruning method for improving model transparency is dataset-dependent and largely empirical and task-dependent.

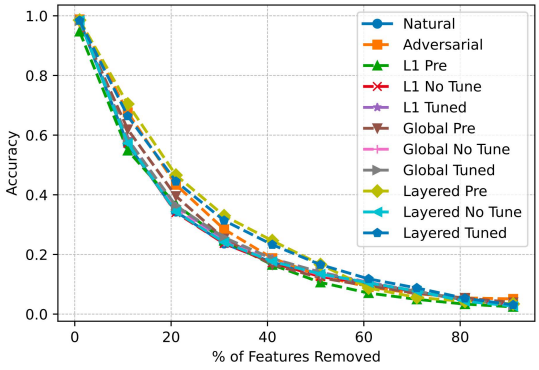

Figure 15: Faithfulness evaluation of SmoothGrad using ROAD for GTSRB using different strategies.

7 Conclusion

This study underscores the critical role of training strategies in shaping the interpretability of AI systems for vehicular applications. By systematically evaluating the impact of natural training, adversarial training, and pruning strategies on the interpretability of deep learning models for traffic sign recognition, we assess how different training strategies affect sparsity, and faithfulness of explanations. Our findings demonstrate that pruning consistently enhances interpretability by generating comprehensible, and highly faithful saliency maps. While the suitable type of pruning is dataset-dependent and requires empirical validation, the results across two different datasets suggest that pruning plays a dual role, improving model efficiency while simultaneously enhancing explanation quality, and practitioners should prioritize pruning-based strategies to improve explanation quality.

Acknowledgment

This work was supported by Toyota InfoTech Labs through Unrestricted Research Funds.

References

[1] Reza Abbasi-Asl and Bin Yu. Interpreting convolutional neural networks through compression. *arXiv preprint arXiv:1711.02329*, 2017.

[2] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *NeurIPS 2018*.

[3] Naman Bansal, Chirag Agarwal, and Anh Nguyen. Sam: The sensitivity of attribution methods to hyperparameters. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 8673–8683, 2020.

[4] Dipkamal Bhusal, Rosalyn Shin, Ajay Ashok Shewale, Monish Kumar Manikya Veerabhadran, Michael Clifford, Sara Rampazzi, and Nidhi Rastogi. Sok: Modeling explainability in security analytics for interpretability, trustworthiness, and usability. In *Proceedings of the 18th International Conference on Availability, Reliability and Security*, pages 1–12, 2023.

[5] Davis Blalock, Jose Javier Gonzalez Ortiz, Jonathan Frankle, and John Guttag. What is the state of neural network pruning? *Proceedings of machine learning and systems*, 2:129–146, 2020.

[6] Kirill Bykov, Anna Hedström, Shinichi Nakajima, and Marina M-C Höhne. Noisegrad—enhancing explanations by introducing stochasticity to model weights. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 6132–6140, 2022.

[7] Prasad Chalasani, Jiefeng Chen, Amrita Roy Chowdhury, Xi Wu, and Somesh Jha. Concise explanations of neural networks using adversarial training. In *International Conference on Machine Learning*, pages 1383–1391. PMLR, 2020.

[8] Hongrong Cheng, Miao Zhang, and Javen Qinfeng Shi. A survey on deep neural network pruning: Taxonomy, comparison, analysis, and recommendations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[9] Xuanyi Dong, Junshi Huang, Yi Yang, and Shuicheng Yan. More is less: A more complicated network with less inference complexity. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5840–5848, 2017.

[10] Christian Etmann, Sebastian Lunz, Peter Maass, and Carola Schoenlieb. On the connection between adversarial robustness and saliency map interpretability. In *International Conference on Machine Learning*, pages 1823–1832. PMLR, 2019.

[11] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

[12] Shizhan Gong, Qi Dou, and Farzan Farnia. Structured gradient-based interpretations via norm-regularized adversarial training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11009–11018, 2024.

[13] Ian Goodfellow. Deep learning, 2016.

[14] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

[15] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

[16] Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28, 2015.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[18] Yang He, Guoliang Kang, Xuanyi Dong, Yanwei Fu, and Yi Yang. Soft filter pruning for accelerating deep convolutional neural networks. *arXiv preprint arXiv:1808.06866*, 2018.

[19] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

[20] Ashkan Khakzar, Soroosh Baselizadeh, Saurabh Khanduja, Christian Rupprecht, Seong Tae Kim, and Nassir Navab. Improving feature attribution through input-specific network pruning. *arXiv preprint arXiv:1911.11081*, 2019.

[21] Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. The (un) reliability of saliency methods. *Explainable AI: Interpreting, explaining and visualizing deep learning*, pages 267–280, 2019.

[22] Sampo Kuutti, Richard Bowden, Yaochu Jin, Phil Barber, and Saber Fallah. A survey of deep learning applications to autonomous vehicle control. *IEEE Transactions on Intelligent Transportation Systems*, 22(2):712–733, 2020.

[23] Ling Liu, Yanzhao Wu, Wenqi Wei, Wenqi Cao, Semih Sahin, and Qi Zhang. Benchmarking Deep Learning Frameworks: Design Considerations, Metrics and Beyond. In *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, pages 1258–1269, July 2018.

[24] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *NeurIPS*, 2017.

[25] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada,*

April 30 - May 3, 2018, Conference Track Proceedings. OpenReview.net, 2018.

[26] Vitali Petsiuk, Abir Das, and Kate Saenko. RISE: randomized input sampling for explanation of black-box models. In *British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018*, page 151. BMVA Press, 2018.

[27] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *22nd ACM SIGKDD*, 2016.

[28] Yao Rong, Tobias Leemann, Vadim Borisov, Gjergji Kasneci, and Enkelejda Kasneci. A consistent and efficient evaluation strategy for attribution methods. In *Proceedings of the 39th International Conference on Machine Learning*, pages 18770–18795. PMLR, 2022.

[29] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[30] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International conference on machine learning*, pages 3145–3153. PMlR, 2017.

[31] Karen Simonyan. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[32] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps, 2014.

[33] Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh, and Himabindu Lakkaraju. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 180–186, 2020.

[34] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

[35] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[36] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, 32:323–332, 2012.

[37] Pascal Sturmfels, Scott Lundberg, and Su-In Lee. Visualizing the impact of feature attribution baselines. *Distill*, 5(1):e22, 2020.

[38] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks, 2017.

[39] Hanxiao Tan. Evaluating explanation robustness to model pruning. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2024.

[40] Chaoqi Wang, Guodong Zhang, and Roger Grosse. Picking winning tickets before training by preserving gradient flow. *arXiv preprint arXiv:2002.07376*, 2020.

[41] David Weber, Florian Merkle, Pascal Schöttle, and Stephan Schlögl. Less is more: The influence of pruning on the explainability of cnns. *arXiv preprint arXiv:2302.08878*, 2023.

[42] Less Wright. Ranger - a synergistic optimizer. https://github.com/lessw2020/Ranger-Deep-Learning-Optimizer, 2019.

[43] Tianyuan Zhang and Zhanxing Zhu. Interpreting adversarially trained convolutional neural networks. In *International Conference on Machine Learning*, pages 7502–7511. PMLR, 2019.

Table 6: Model performance of natural training (Nat), adversarial training (Adv), unstructured L1 pruning (L1), global pruning (Global), and layered structured pruning (Layered) on ResNet-based model for LISA dataset. Here, Pre-train, and Post-train means pruning before training and after training. FT means fine-tuning.

Model	Nat	Adv	L1	Global	Layered
Pre-train	98.3	84.2	97.1	96.1	95.1
Post-train (FT)	98.3	84.2	96.8	98.8	99.4
Post-train (no FT)	98.3	84.2	97.4	97.9	89.1

A LISA dataset - ResNet network

Table 6 shows the model performance on different training strategy with ResNet network on LISA dataset. We can clearly observe that adversarial training compromises benign performance significantly compared to the pruned models.

Figures 16, 17 and 18 show the saliency maps for different training strategies on ResNet network, revealing sparse and comprehensible saliency maps for pruned models. As before, Integrated Gradients create much cleaner saliency maps. However, in Vanilla Gradient and SmoothGrad, we can observe less noisy saliency maps with pruned models.

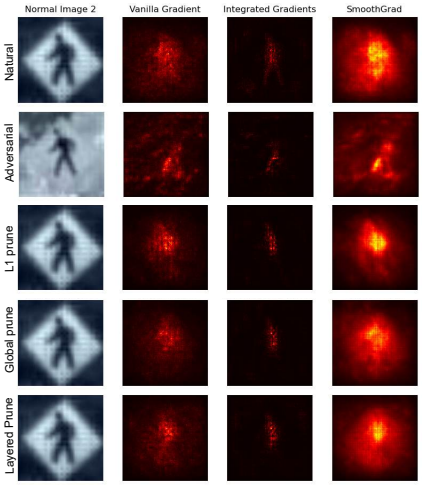

Figure 16: Saliency maps comparison for LISA dataset between natural training, adversarial training and pre-train pruning for ResNet model.

Table 7 presents a comparative evaluation of sparsity scores. As before, there is no significant gain in sparsity with pruned models. However, the noisy saliency maps in adversarially trained models are validated by negative scores in the table.

Figure 19 shows the faithfulness evaluation for Vanilla Gradient method, where adversarial trained model has sharper drop in accuracy compared to other approaches especially up-to feature removal of 50%. All other pruning approaches

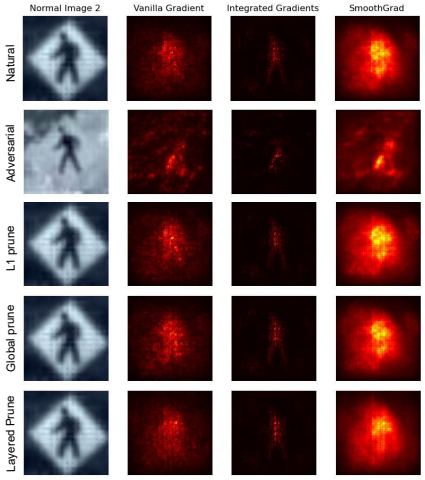

Figure 17: Saliency maps comparison for LISA dataset between natural training, adversarial training and test-time pruning with no fine-tuning for ResNet model.

however surpass adversarial training method following this. As seen in the plot, the naturally trained model has almost a flat curve, signifying that the explanations produced from naturally trained model are not faithful to the model. This raises questions on the the quality of saliency maps using Vanilla Gradient (VG) with the naturally trained model as they do not truly reflect the underlying model.

Similar observations can be made from Figure 20 that shows the faithfulness evaluation for Integrated Gradient method. The results demonstrate that pruning generally enhances explanation faithfulness, as indicated by a sharper drop in accuracy when the most relevant features are removed. This is prominent with layered (no fine tune method). As more features are removed, layered (with fine tune) shows the sharpest decline in accuracy, confirming its faithfulness. In comparison, naturally trained model has a flat line, confirming that natural training does not lead to faithful explanations.

Figure 21 that shows the faithfulness evaluation for SmoothGrad further validates that pruning enhances the faithfulness of saliency maps, as models trained with pruning generally exhibit a steeper accuracy decline when key features are removed. As observed in the figure, Layered Pre, L1 Pre and Global Pre (pre-train pruning) have the steepest decline demonstrating that pruning significantly improves the faithfulness of explanations.

B GTSRB Dataset: Post-Train Prune

Figure 22 and Figure 23 shows the saliency maps comparison between naturally trained, adversarially trained and pruned models without and with fine-tuning respectively.

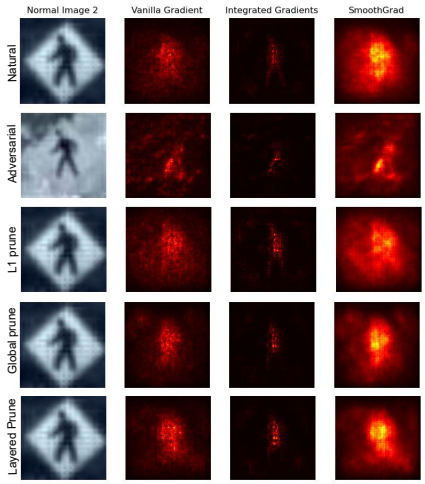

Figure 18: Saliency maps comparison for LISA dataset between natural training, adversarial training and test-time pruning with fine-tuning for ResNet model.

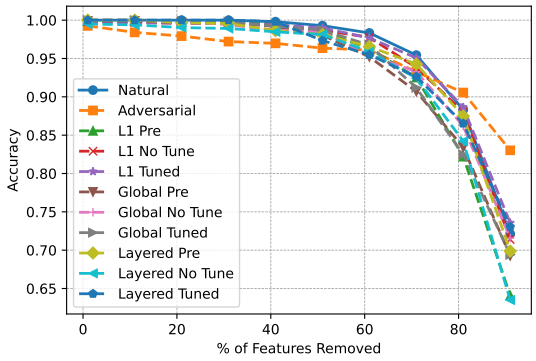

Figure 19: Faithfulness evaluation of Vanilla Gradient using ROAD on ResNET using different strategies.

Table 7: Sparsity evaluation of Vanilla Gradient (VG), Integrated Gradient (IG) and SmoothGrad (SG) on different training strategy of ResNet model for LISA dataset. Here, adv means adversarial training and L1, Global and Layered means L1 unstructured pruning, Global pruning and layered structured pruning. Pre-train, and Post-train means pruning before training and after training. FT means fine-tuning. Higher the better scores.

Method	Model	Adv	L1	Global	Layered
VG	Pre-train	-0.03	0.01	0.00	-0.01
	Post-train (FT)	-0.03	0.00	0.00	-0.02
	Post-train (no FT)	-0.03	-0.01	0.02	0.00
IG	Pre-train	-0.01	0.01	0.00	0.00
	Post-train (FT)	-0.01	0.00	0.02	0.00
	Post-train (no FT)	-0.01	0.00	0.00	-0.01
SG	Pre-train	-0.06	0.02	0.00	-0.01
	Post-train (FT)	-0.06	-0.01	0.03	0.00
	Post-train (no FT)	-0.06	0.00	0.00	-0.02

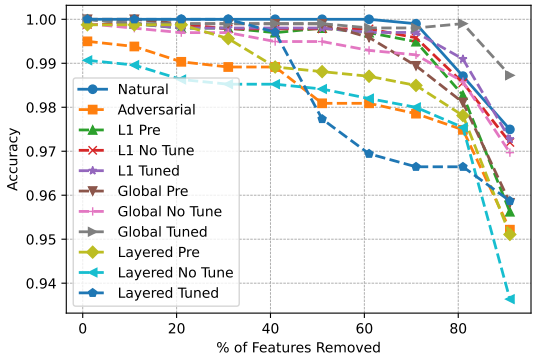

Figure 20: Faithfulness evaluation of Integrated Gradient using ROAD on ResNet using different strategies.

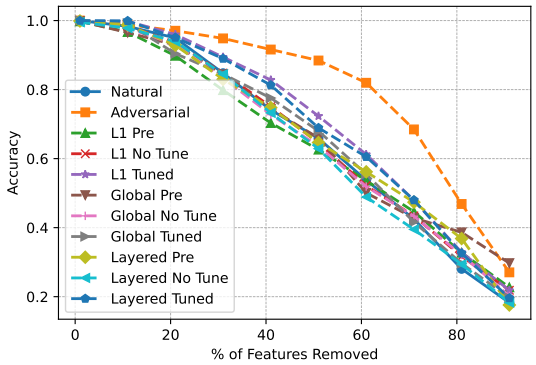

Figure 21: Faithfulness evaluation of Smooth Gradient using ROAD on ResNet using different strategies.

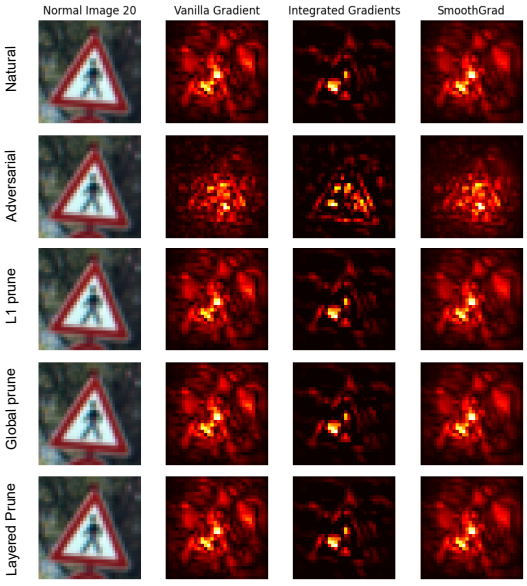

Figure 22: Saliency maps comparison for GTSRB dataset between natural training, adversarial training and post-train pruning without fine-tuning.

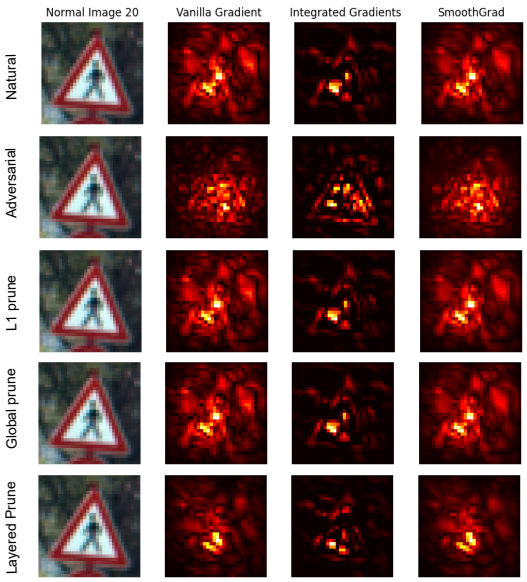

Figure 23: Saliency maps comparison for GTSRB dataset between natural training, adversarial training and post-train pruning with fine-tuning.