Generalization of Interpretable Deep Learning Requires More Data

Ching Lam Choi*, Farzan Farnia†

Abstract

Feature saliency maps are commonly used for interpreting neural network predictions. This approach to interpretability is often studied as a post-processing problem independent of training setups, where the gradients of trained models are used to explain their output predictions. However, in this work, we observe that gradient-based interpretation methods are highly sensitive to the training set: models trained on disjoint datasets without regularization produce inconsistent interpretations across test data. Our numerical observations pose the question of how many training samples are required for accurate gradient-based interpretations. To address this question, we study the generalization aspect of gradient-based explanation schemes and show that the proper generalization of interpretations from training samples to test data requires more training data than standard deep supervised learning problems. We prove generalization error bounds for widely-used gradient-based interpretations, suggesting that the sample complexity of interpretable deep learning is greater than that of standard deep learning. Our bounds also indicate that Gaussian smoothing in the widely-used SmoothGrad method plays the role of a regularization mechanism for reducing the generalization gap. We evaluate our findings on various neural net architectures and datasets, to shed light on how training data affect the generalization of interpretation methods.

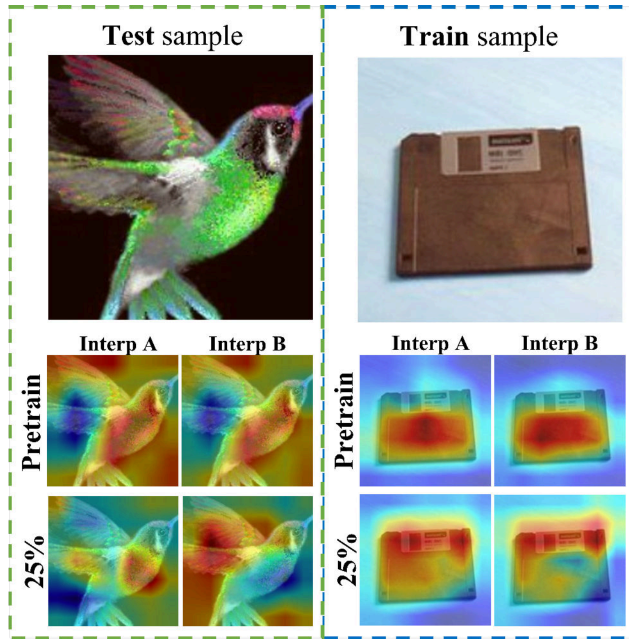


Figure 1: Grad-CAM interpretations across ResNets on Caltech-256. Each pair of models is trained over the train set split factor of $sf = 4$ (25% of training data), with and without pre-training on ImageNet.

1. Introduction

Multi-layer neural network (NN) models have achieved revolutionary success in computer vision problems including image recognition [17], object detection [41], and medical image processing [21]. This success is primarily due to the enormous capacity of NNs as well as their impressive generalization performance from training samples to unseen data. In other words, not only do massive NNs perform almost perfectly in predicting the label of training samples, but also they maintain their satisfactory training performance on test data unobserved during the NN model's training. The mysterious generalization success of deep learning models has attracted a lot of attention in the machine learning community.

While NNs achieve great prediction performance over standard computer vision datasets, their deployment in real-world applications such as self-driving cars and machine-based medical diagnostics requires a reliable interpretation of their predictions. Such interpretation of these large-scale models will help domain experts understand the basis of their predictions to further improve and robustify the prediction model. Over the recent years, several algorithms have been developed to give such an interpretation, including the widely-used gradient-based feature saliency maps such

*Department of Computer Science and Engineering, The Chinese University of Hong Kong, clchoi1@cse.cuhk.edu.hk

†Department of Computer Science and Engineering, The Chinese University of Hong Kong, farnia@cse.cuhk.edu.hk

as the simple gradient [1, 31], integrated gradients [36], and SmoothGrad [32] methods. These gradient-based algorithms are based on the first-order derivative of the NN model's score function with respect to the input variables, which reveal the features with a major impact on the model's prediction.

While the gradient-based interpretation methods have found many applications in computer vision problems, the theoretical understanding of the underlying factors contributing to their performance is still largely inadequate. Specifically, the generalization aspect of standard interpretation methods has not been studied in the literature, and it remains unclear how many training samples are required to obtain an accurate estimation of the gradient-based explanation. Characterizing the sample complexity of learning generalizable saliency maps is an important step toward understanding the fundamental limits of interpreting NN models and developing effective regularization schemes for improving their performance.

In this paper, we focus on the generalization aspect of the deep learning-based interpretation methods, and through several theoretical and numerical results attempt to show that the proper generalization of a NN's saliency map requires a larger training set than the standard classification problem focusing only on the accuracy of the prediction model. In other words, the sample complexity of finding an interpretable and accurate deep learning model is greater than that of training an only accurate NN classifier.

To support the above statement on the generalization of interpretable deep learning, we prove theoretical bounds on the generalization rate of standard gradient-based saliency maps, including simple and integrated gradients, from training samples to test data. Our generalization bounds indicate the considerable discrepancy between the training and test performance scores of gradient-based interpretation schemes. We compare the shown generalization error bounds with the standard bounds on the generalization error of multi-layer NN classifiers, which suggests a higher statistical complexity for the interpretation of neural nets than for the accuracy of a NN classifier as characterized by [2].

Subsequently, we focus on the SmoothGrad algorithm and show that the Gaussian smoothing in this method can be interpreted as a regularization mechanism controlling the difference between test and training interpretation performance. Our results indicate that the generalization error will decrease linearly with the standard deviation of the SmoothGrad noise, which will reduce the variance of the saliency map at the cost of a higher bias toward a constant interpretation. Therefore, this result would parallel the well-known bias-variance trade-off for norm-based regularization methods in the context of supervised learning.

Finally, we present the results of several numerical experiments demonstrating the effect of the number of training data on the variance of the gradient-based saliency maps. Our empirical findings reveal the significant impact of the size of the training set on the estimated saliency map for unseen test data. We show that standard methods such as simple and integrated gradients are highly susceptible to the samples in the training set. In addition, our results show a lower correlation between gradient-based interpretation maps of two NNs with disjoint training sets than the correlation between the NNs' predicted labels, indicating that an interpretable NN model demands more training data than an accurate NN classifier.

On the other hand, we numerically show the regularization effect of the SmoothGrad algorithm which manages to properly control the variance of the saliency map on test data. Our numerical results indicate the importance of proper generalization in the visual performance of interpretation methods and support the SmoothGrad approach as a regularized interpretation scheme. For example, Fig. 1 shows two samples in the Caltech-256 dataset, each interpreted with two pairs of different ResNet-50s: one pair trained over 25% of the training set plus pre-training on ImageNet, and the other pair only trained over 25% of the training set. The heatmap discrepancy is visible in the 25% case and more significant for the test sample, whereas for the full training set plus pertaining, the discrepancy is negligible. Here, we summarize our contributions:

- Highlighting the role of generalization in the performance of deep learning interpretation methods,
- Proving theoretical generalization bounds for standard gradient-based saliency maps,
- Demonstrating the regularization effect of Gaussian smoothing in the SmoothGrad approach,
- Providing numerical results on the generalization of interpretations and regularization in SmoothGrad.

2. Related Work

Standard generalization analysis in deep learning focuses on the consistency of neural nets' predictions across training and test samples. However, neural nets have been shown to memorize random labels and Gaussian pixel inputs [40]; to easily overfit dataset biases and labeling errors [34, 3, 29], generating unexplainable predictions and exhibiting weak classification decision boundaries. To debug these faulty predictions, several post-hoc interpretability [20] methods attempt to explain the outputs via visualizations, counterfactuals and numerical metrics. Unlike multi-modal concept learning methods, such as TCAV [13], Concept Bottleneck Models (CBM) [14] and Interpretable Basis Decomposition (IBD) [43], post-hoc methods study interpretations as a stand-alone problem independent of the blackbox model training process and setup.In this work, we choose a different approach by experimenting on different gradient-based and feature-based methods, to show that the training-to-test generalization of interpretations depends heavily on the training set size.

Gradient-based Interpretations. Gradients of the model output with respect to its input is an intuitive way of attributing the prediction to the data representation [36]. Early attribution techniques generate explanations from the product between simple gradients and features [1, 31]; works such as Guided BackProp [33], DeConvNet [39], DeepLift [30] and Layer-wise Relevance Propagation (LRP) [5] utilize discrete step backpropagation to proportionally attribute class-wise prediction scores to network features. Sundararajan *et al.* [36] further improve the reliability of using gradients to weigh feature importance, by proposing integrated gradients to satisfy desirable axioms of sensitivity and implementation invariance.

Gradients can also measure interpretability within and between trained models. Gradient signal-to-noise ratio (GSNR) [22] uses gradient alignment across different samples to understand representation consistency of a model; Raghu *et al.* [26] utilize the norm of network gradients to quantify the amount of discrepancy between the input and prediction. The difference of gradients between 2 networks taken with respect to the same input evaluates how much the networks' predictions disagree. In this work, we experiment on gradient-based feature attribution methods of simple gradients and integrated gradients. We further calculate the Frobenius norm and distance of networks' gradient interpretations to interpret prediction consistency and agreement.

Parameter Space Interpretations. Beyond gradient-based analysis, the representation similarity of samples between networks and network layers is also an important interpretation metric. Class Activation Mapping (CAM) [42] and the subsequent Grad-CAM [28] utilize inherent localization properties of deep NN features to visualize salient regions in images. They project the target class' weights from the output layer back to the convolutional feature maps, using network parameter activations to score the importance of image features for classification. By comparing the CAM interpretations of trained models, we qualitatively assess how consistently do they attend to the same spatial regions. To directly compare between networks and across layers, Centered Kernel Alignment (CKA) [15] improved upon canonical correlation analysis methods [25, 24], by calculating the similarity index between representational matrices. Their results generalize to different kernels, network architectures and layer types, providing us with insight into the similarity between differently trained models, across layers and samples.

Robustness and Consistency of Interpretations. Several related papers analyze the fragility and consistency of the standard saliency maps. The related papers [9, 6, 12, 35] show that standard gradient-based interpretations of neural nets commonly lack robustness to input perturbations, and the manipulated interpretation can transfer across neural net architectures. Levine *et al.* [19] present a certifiably robust interpretation scheme by applying sparsification to the SmoothGrad approach. In another related paper, Fel *et al.* [8] analyze the consistency and algorithmic stability of standard interpretation methods and measure the sensitivity of interpretation methods to the inclusion of one specific sample in the training set. However, unlike our work, the mentioned works do not focus on the generalization of interpretation methods from training to test data.

3. Preliminaries

In this section, we discuss the notation and definitions used throughout the paper and shortly review the gradient-based saliency maps analyzed in the paper.

3.1. Notation

In the paper, we use notation $\mathbf{X} \in \mathbb{R}^d$ to denote the random feature vector and $Y \in \{1, \dots, k\}$ to denote the k-ary classification label. The deep learning algorithm trains a neural network $f_{\mathbf{w}} \in \mathcal{F}$ where $\mathbf{w}$ represents the vector containing the weights of the neural net function and $\mathcal{F} = \{f_{\mathbf{w}} : \mathbf{w} \in \mathcal{W}\}$ denotes the feasible set of functions including the neural nets with allowed weight vectors in set $\mathcal{W}$. Note that every $f_{\mathbf{w}} : \mathbb{R}^d \to \mathbb{R}^k$ maps the d-dimensional input to a k-dimensional prediction vector including a real-valued entry for every label.

For training the neural net, we follow the standard empirical risk minimization (ERM) method minimizing the empirical expected loss, measured with loss function $\ell(\hat{y}, y)$ between actual y and predicted $\hat{y}$ labels, over the training set $\{(\mathbf{x}_i, y_i)_{i=1}^n\}$ consisting of n labeled training examples drawn independently from an underlying distribution $P_{\mathbf{X},Y}$:

$$\min_{\mathbf{w}\in\mathcal{W}} \frac{1}{n}\sum_{i=1}^{n} \ell\bigl(f_{\mathbf{w}}(\mathbf{x}_i), y_i\bigr). \tag{1}$$

We note that the standard generalization analysis in machine learning focuses on the difference between the expected loss values on the training samples and the test samples drawn from the underlying model $P_{\mathbf{X},Y}$.

3.2. Gradient-based Saliency Maps

In our generalization analysis, we consider standard gradient-based saliency maps as a neural net's interpretation. To define standard saliency maps, we use $f_c(\mathbf{x})$ to denote the real-valued output of the c-th neuron at the final layer of neural net f. Assuming that c is the assigned label to input $\mathbf{x}$, i.e. the final layer's neuron with the maximum value, we review the definitions of the following standard saliency maps:

1. **Simple Gradient Method**: As defined by Simonyan *et al.* [31], the simple gradient is the gradient of the neural net's output at the predicted neuron with respect to the input feature vector:

$$\text{Simple-Grad}(\mathbf{f}_c, \mathbf{x}) := \nabla_{\mathbf{x}} \mathbf{f}_c(\mathbf{x}). \tag{2}$$

2. **Integrated Gradients**: Given a reference vector $\mathbf{x}^0$, the integrated gradients [36] calculate the gradient's integral over the line segment connecting the reference point $\mathbf{x}^0$ and a target point $\mathbf{x}$. In practice, the integrated gradient is approximated using m intermediate points between $\mathbf{x}^0$ and $\mathbf{x}$:

$$\begin{aligned}\text{Int-Grad}(\mathbf{f}_c, \mathbf{x}) &:= \int_0^1 \nabla_{\mathbf{x}} \mathbf{f}_c\big(\mathbf{x}^0 + \alpha \Delta\mathbf{x}\big) \odot \Delta\mathbf{x}\, \mathrm{d}\alpha \\ &\approx \frac{\Delta\mathbf{x}}{m} \odot \sum_{i=1}^{m} \nabla_{\mathbf{x}} \mathbf{f}_c\big(\mathbf{x}^0 + \frac{i}{m}\Delta\mathbf{x}\big).\end{aligned}$$

In the above, $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}^0$ denotes the difference between the reference and target points, and $\odot$ denotes the vector element-wise product.

3. **SmoothGrad**: The SmoothGrad approach [32] applies Gaussian smoothing to the gradient-based interpretation, and calculates the average gradient with an isotropic Gaussian distribution centered at the target data point $\mathbf{x}$. Specifically, we define Gaussian vector $\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \sigma^2 I)$ and define SmoothGrad as

$$\begin{aligned}\text{Smooth-Grad}(\mathbf{f}_c, \mathbf{x}) &:= \mathbb{E}_{\mathbf{Z}\sim\mathcal{N}(\mathbf{0},\sigma^2 I)}\big[\nabla f_c(\mathbf{x} + \mathbf{Z})\big] \\ &\approx \frac{1}{m}\sum_{i=1}^{m} \nabla f_c(\mathbf{x} + \mathbf{z}_i),\end{aligned} \tag{3}$$

where $\mathbf{z}_1, \ldots, \mathbf{z}_m \sim \mathcal{N}(\mathbf{0}, \sigma^2 I)$ are independent observations of the Gaussian noise used to approximate the SmoothGrad expectation.

4. Generalization in Interpretation Tasks

Generalization from training examples to test data is a crucial factor behind the success of every learning algorithm. In the case of interpretation methods, we note that the trained neural net $f_{\mathbf{w}} \in \mathcal{F}$ is learned using the training data, and hence the learned function will be different from the optimal neural net minimizing the expected loss over the underlying distribution of test data $P_{\mathbf{X},Y}$. In our discussion, we use f^* to denote the optimal neural net classifier in $\mathcal{F}$ in terms of the achieved performance on test data, i.e.

$$f^* \;:=\; \underset{f\in\mathcal{F}}{\operatorname{argmin}}\; \mathbb{E}_{(\mathbf{X},Y)\sim P}\big[\,\ell\big(f(\mathbf{X}), Y\big)\,\big]. \tag{4}$$

While we, as the learner, do not know the underlying distribution $P_{\mathbf{X},Y}$ and therefore the optimal f^*, we can still define the loss of an interpretation scheme $I(\cdot)$ at an input $\mathbf{x}$ as the norm difference between I's output for a given classifier f and the optimal f^*, that is

$$\text{Loss}_I(f, \mathbf{x}) \;:=\; \big\| I(f, \mathbf{x}) - I(f^*, \mathbf{x}) \big\|_2, \tag{5}$$

where $\|\cdot\|_2$ denotes the L_2-norm of an input vector. Here we define the interpretation vector $I(f, \mathbf{x})$ when we choose class $c = y$ for the actual label y of sample $\mathbf{x}$. Also, note that the above definition uses $I(f^*, \mathbf{x})$ as the underlying interpretation which the learner aims to estimate from training data.

Definition 1. *For a classifier function f and training set $\{(\mathbf{x}_i, y_i)_{i=1}^n\}$, we define the interpretation training loss $\widehat{\mathcal{L}}(f)$ as the expected interpretation loss on training data:*

$$\widehat{\mathcal{L}}(f) \;:=\; \frac{1}{n}\sum_{i=1}^{n} \text{Loss}_I(f, \mathbf{x}_i).$$

Also, we define the interpretation test loss $\mathcal{L}(f)$ as the expected interpretation loss on the underlying distribution of test data $P_{\mathbf{X}}$:

$$\mathcal{L}(f) \;:=\; \mathbb{E}_{\mathbf{X}\sim P_{\mathbf{X}}}\big[\,\text{Loss}_I(f, \mathbf{X})\,\big].$$

Finally, we define the interpretation generalization error as the difference between the interpretation training and test loss values:

$$\epsilon_{\text{gen}}(f) := \mathcal{L}(f) - \widehat{\mathcal{L}}(f).$$

Based on the above definition, a necessary condition for a reliable interpretation of a neural network's prediction is a small interpretation generalization error. This condition is required, because if the generalization error is relatively large, then the accuracy of the interpretation scheme will be significantly worse on test data than on the training samples. Note that while the generalization condition is necessary for a proper interpretation result on test samples, it is still not sufficient for a satisfactory interpretation performance, since it also requires a good performance on training data. In the next section, we present theoretical bounds on the interpretation generalization error of neural network classifiers, to compare the generalization rates across standard and interpretable deep learning problems.

5. Theoretical bounds on Interpretation Generalization Error

In this section, we theoretically analyze the interpretation generalization error of neural networks. Here we suppose that the neural net function $f_{\mathbf{w}}:\mathbb{R}^d\rightarrow\mathbb{R}^k$ has the following format:

$$f_{\mathbf{w}}(\mathbf{x}) = W_L\phi_{L-1}\big(W_{L-1}\phi_{L-2}(\cdots W_2\phi_1(W_1\mathbf{x}))\cdot\big). \quad (6)$$

Here the vector $\mathbf{w}$ concatenates the entries of the L layers' weight matrices $W_1,\ldots,W_L$. Also, $\phi_i:\mathbb{R}\rightarrow\mathbb{R}$ represents the activation function at layer i.

Our first theorem concerns the interpretation generalization performance of the simple gradient and integrated gradients. This result demonstrates that the generalization of these gradient-based interpretation schemes could require a larger training set than the standard deep learning classification problem. Specifically, this theorem extends the generalization analysis in [2] to the gradient-based interpretation of neural networks. In the following, we use $\|\cdot\|_2$ to denote a matrix's spectral norm, i.e. its largest singular value, and also $\|\cdot\|_{2,1}$ denotes the $L_{2,1}$-group norm of a matrix, i.e. the summation of the L_2-norms of the matrix's rows.

Theorem 1. *Suppose that the neural net classifier in (6) has an γ_i-Lipschitz and γ_i-smooth activation function satisfying $\forall z\in\mathbb{R}:\max\{|\phi_i'(z)|,|\phi_i''(z)|\}\le\gamma_i$. We assume that the interpretation loss is upper-bounded by constant c and the training data matrix $\mathbf{X}_{n\times d}$ is norm-bounded as $\|\mathbf{X}\|_2\le B$ with probability 1. Also, we use D to denote the maximum number of rows and columns in $f_{\mathbf{w}}$'s weight matrices. Then, for every $\omega>0$, with probability at least $1-\omega$ the following generalization error bound will hold for both the simple gradient method and integrated gradients of every $f_{\mathbf{w}}$:*

$$\epsilon_{\mathrm{gen}}(f_{\mathbf{w}})\le\mathcal{O}\Big(c\sqrt{\frac{\log(1/\omega)}{n}}+\frac{BR_{\mathbf{w}}\log(n)\log(D)}{n}\Big).$$

Here $R_{\mathbf{w}}:=\big(\sum_{i=1}^L\prod_{j=1}^i\gamma_j\|\mathbf{W}_j\|_2\big)\big(\prod_{i=1}^L\gamma_i\|\mathbf{W}_i\|_2\big)\times\big(\sum_{i=1}^L\frac{\|W_i\|_{2,1}^{2/3}}{\|W_i\|_2^{2/3}}\big)^{3/2}$ denotes the interpretation capacity of the neural net.

Proof. We present the proof in the Appendix. □

Comparing the generalization bound for the simple and integrated gradients interpretation to the generalization bound in [2] for the standard supervised learning task, we notice an order-wise $\mathcal{O}\big(\sum_{i=1}^L\prod_{j=1}^i\gamma_j\|\mathbf{W}_j\|_2\big)$ greater generalization error for gradient-based interpretation schemes. This additional term indicates the extra cost of generalization for the simple and integrated gradients-based interpretable deep learning. Next, we state the generalization bound for the SmoothGrad approach.

Theorem 2. *Suppose that the neural net classifier in (6) has an γ_i-Lipschitz activation function satisfying $\forall z\in\mathbb{R}:\ |\phi_i'(z)|\le\gamma_i$. We assume that the interpretation loss is upper-bounded by constant c and the training data matrix $\mathbf{X}_{n\times d}$ is norm-bounded as $\|\mathbf{X}\|_2\le B$ with probability 1. Then, for every $\omega>0$, with probability at least $1-\omega$ the following generalization error bound will hold for the SmoothGrad interpretation of every $f_{\mathbf{w}}$ with standard deviation $\sigma>0$:*

$$\epsilon_{\mathrm{gen}}(f_{\mathbf{w}})\le\mathcal{O}\Big(c\sqrt{\frac{\log(1/\omega)}{n}}+\frac{BL_{\mathbf{w}}\log(n)\log(D)\sqrt{d}}{n\sigma}\Big),$$

where $L_{\mathbf{w}}:=\prod_{i=1}^L\gamma_i\|\mathbf{W}_i\|_2\big(\sum_{i=1}^L\frac{\|W_i\|_{2,1}^{2/3}}{\|W_i\|_2^{2/3}}\big)^{3/2}$ denotes the spectral capacity of the neural network.

Proof. We present the proof in the Appendix. □

Note that Theorem 2's bound is only by a multiplicative factor $\frac{\sqrt{d}}{\sigma}$ different from the generalization bound in the standard deep supervised learning problem [2]. Therefore, the theorem suggests that Gaussian smoothing can be interpreted as a regularization of the simple gradient approach to improve its generalization behavior. The SmoothGrad interpretation algorithm could gain a better generalization performance by increasing the standard deviation, while the training performance could drop because of the additional noise.

6. Numerical Experiments

6.1. Experimental Details

Datasets. We numerically study the generalization and visual consistency of interpretation methods on the standard CIFAR-10 [16] and the larger scale Tiny-ImageNet [18] and Caltech-256 [10] datasets. Tiny-ImageNet dataset is a downsampled subset of ImageNet [27] and comprises 200 object categories with 500 training images and 50 validation images for each class. Caltech-256 contains 256 object categories totaling 30,607 high-resolution images. We note that since

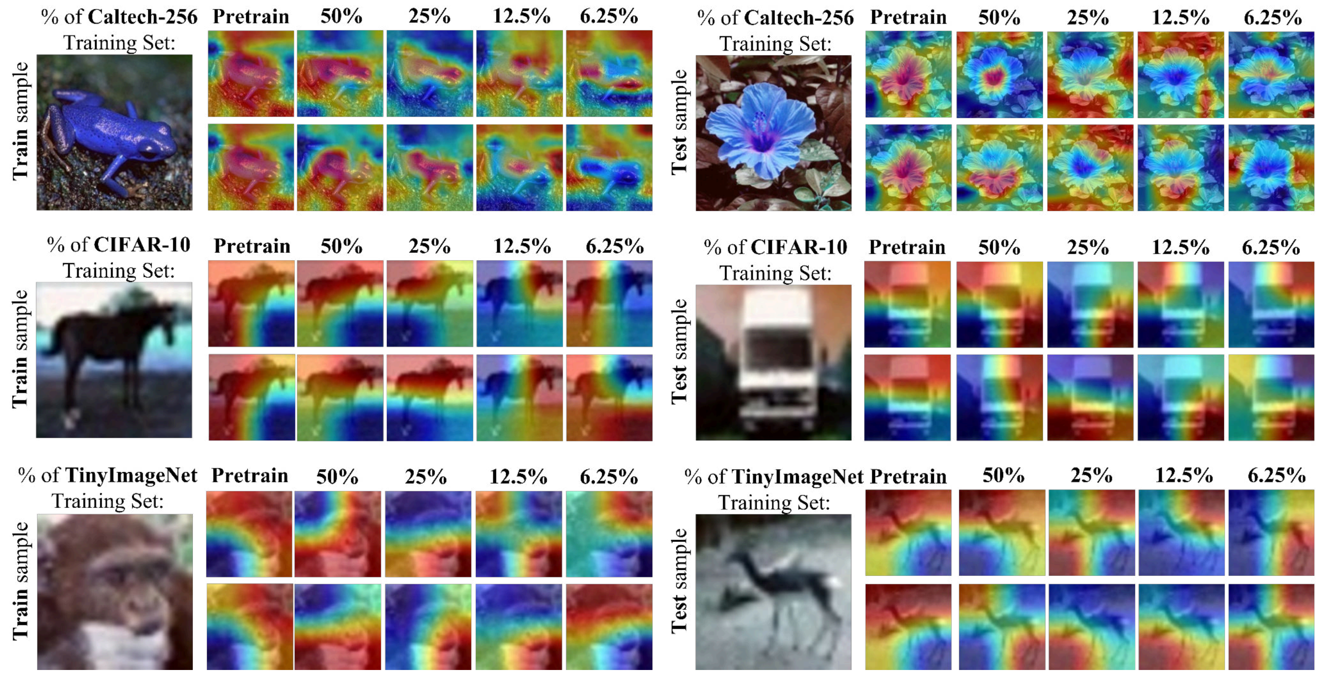


Figure 2: Grad-CAM comparisons with ConvNeXt-Tiny. As we increase the number of training samples from 6.25% ($sf = 16$) of the training set, to using 50% of the training set, then to pre-training on ImageNet plus fine-tuning with 50% training data, we observe that model pairs generate increasingly consistent interpretations.

our experiments would require us to train from scratch a multitude networks on different subset levels for each dataset, it was infeasible to directly experiment on the large-scale ImageNet [27] dataset. Instead, to validate the message that the generalization of interpretations requires more data, we utilize the large-scale ImageNet dataset for pre-training via off-the-shelf weights.

Neural network architectures. To validate our hypotheses, we experiment on a diverse set of computer vision network architectures. We report our numerical results for the following convolutional neural networks: ConvNeXt-Tiny [23], EfficientNet-V2-S [37], ResNet [11] (we trained ResNet-50 on Caltech-256 and trained ResNet-18 on TinyImageNet, CIFAR-10); for the ViT-B-16 Vision Transformer model proposed by Dosovitskiy *et al.* [7, 4]; and for the multi-layer perceptron model of MLP-Mixer [38].

Experiment design. To evaluate the effect of training set size on interpretation generalization, we consider split factors of $\mathrm{sf} = 2, 4, 8, 16$, each corresponding to training with $50\%, 25\%, 12.5\%, 6.25\%$ of available training data. We train a neural net for every data subset for 200 epochs. To further improve the interpretation generalization, we allow models to train on "more data" by using pre-trained ImageNet weights, then fine-tuning for 50 epochs.

6.2. Verifying the Generalization Gap

We visualize the interpretation generalization gap in Fig. 2, by varying the number of training samples from 6.25% of the training set, to pre-training on ImageNet and fine-tuning on 50% of the train set. As the number of training samples increased, GradCAM [28] interpretations became more similar between pairs of models, as seen from how "Pretrain" model pairs have near-perfect saliency map agreement across datasets. For models that are optimized with more training samples, this localization ability transfers successfully to unseen test data, verifying that more training samples are required for interpretations to agree across models and generalize across train and test sets.

6.3. Gradient-based interpretations

In Figure 3, through qualitative experiments on Caltech-256 [10] with the simple gradient [31], Smooth-Grad [32], integrated gradients [36] and DeepLift [30], we show that "Pretrain" models outperform 6.25% models in terms of visual fidelity, localization meaningfulness and generalization ability to test samples.

We present further numerical evidence by assessing the generalization gap in *integrated gradients* [36]. We vary the dataset split factor from 2 ($\frac{1}{2}$ of train set), 4, 8 to 16 ($\frac{1}{16}$ of the train set) and generated mass-

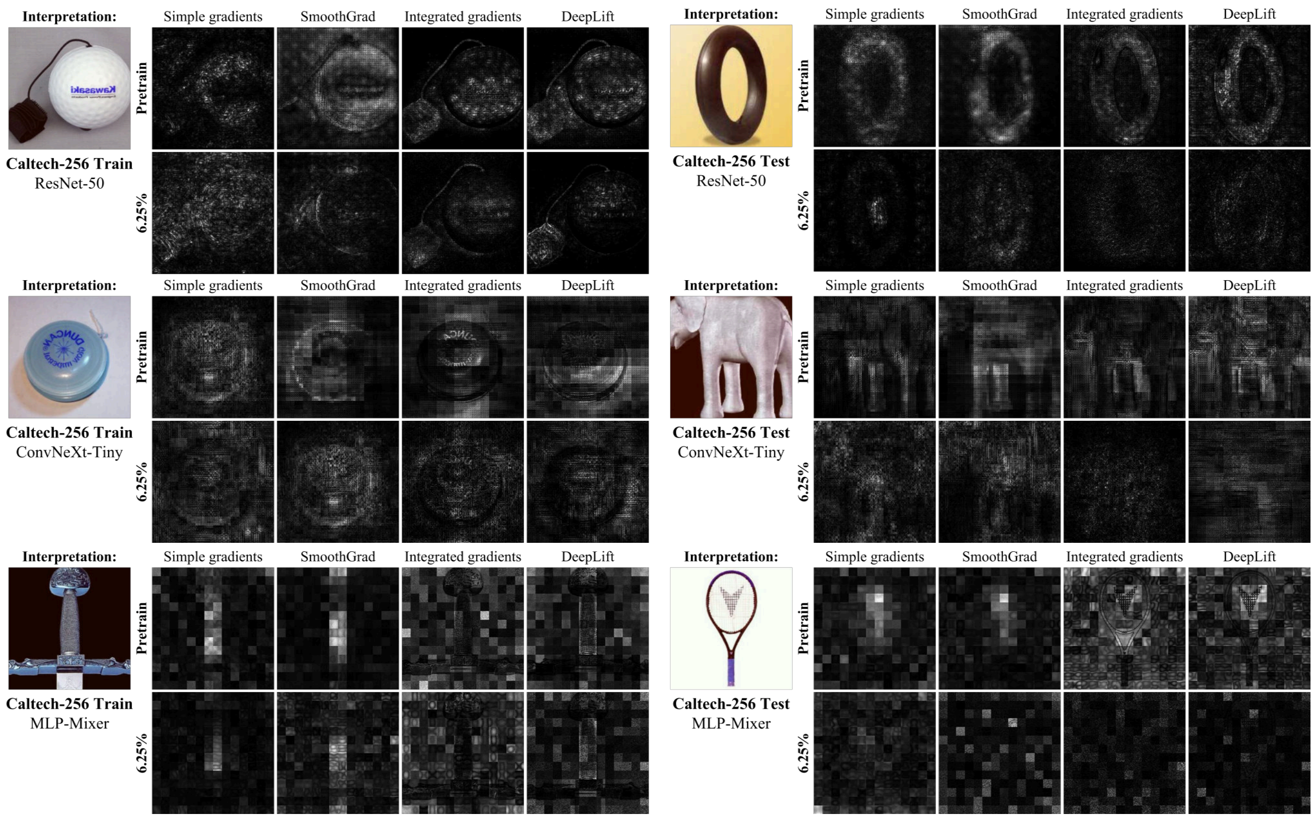
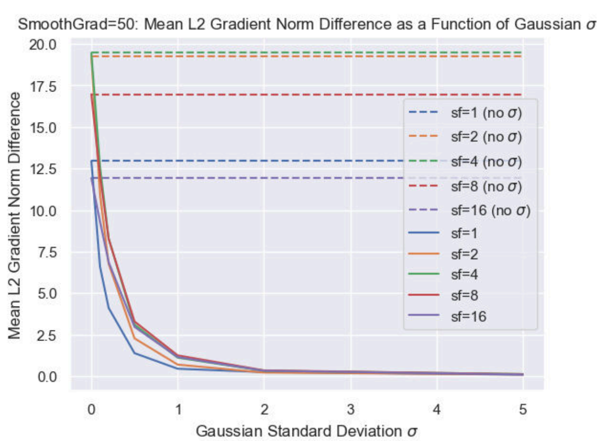
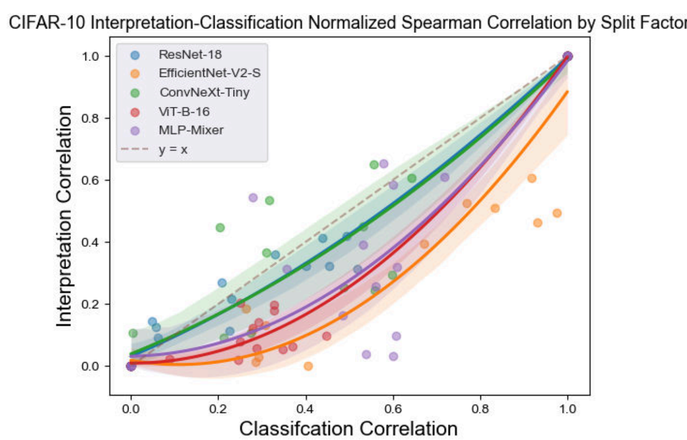


Figure 3: Different gradient-based interpretation methods tested on Caltech-256. We compare the fidelity, localization meaningfulness and train-test generalization abilities of interpretations, for Pretrain and the 6.25% experimental settings. We observe that the generalization and performance gaps widen for interpretations generated by models trained on smaller, disjoint training sets. Full results are in the Appendix.



(a) ResNet-18 on CIFAR-10 gradient norm difference.



(b) CIFAR-10 interpretation-classification correlation scores.

Figure 4: 4a) Averaged L_2-norm difference between gradient interpretations of neural nets trained with a split factor of sf $= 1, 2, 4, 8, 16$ with and without (no σ) SmoothGrad. 4b) Normalized Spearman correlation of network interpretations against softmax predictions. As sf increases from 2 to 16 and models are trained with smaller disjoint train sets, the rank correlation of test set interpretations drop more acutely than that of network predictions. Results of other datasets are in the Appendix.

Rank Correlation & Saliency Pixel Intersection by Split Factor							
Dataset / **Model**		ResNet-18/50		ViT-B-16		MLP-Mixer	
		Rank C ↑	Px % ↑	Rank C ↑	Px % ↑	Rank C ↑	Px % ↑
CIFAR-10	$sf = 2$	**.37** ± .02	**28.0** ± 0.9	**.31** ± .02	**23.7** ± 1.6	**.39** ± .01	**34.1** ± 1.3
	$sf = 4$	.33 ± .01	26.8 ± 1.0	.25 ± .02	18.4 ± 1.5	.38 ± .01	33.6 ± 1.0
	$sf = 8$	.31 ± .01	25.3 ± 0.7	.25 ± .02	17.7 ± 1.3	.36 ± .01	32.7 ± 0.9
	$sf = 16$	.28 ± .01	23.8 ± 0.5	.23 ± .02	15.4 ± 1.4	.34 ± .01	28.4 ± 0.9
Caltech-256	$sf = 2$	**.31** ± .01	**3.3** ± 0.1	**.21** ± .05	**3.8** ± 1.4	**.31** ± .01	**4.3** ± 0.3
	$sf = 4$	.30 ± .02	2.2 ± 0.3	.18 ± .05	1.7 ± 0.5	.21 ± .05	1.5 ± 0.4
	$sf = 8$	.27 ± .04	1.7 ± 0.6	.17 ± .02	0.7 ± 0.6	.18 ± .03	1.2 ± 0.7
	$sf = 16$	.24 ± .03	0.1 ± 0.4	.15 ± .03	0.5 ± 0.6	.14 ± .02	0.9 ± 0.7
TinyImageNet	$sf = 2$	**.12** ± .02	**23.8** ± 0.9	**.11** ± .01	**20.3** ± 0.2	**.11** ± .01	**26.3** ± 0.3
	$sf = 4$	.10 ± .03	23.7 ± 0.5	.10 ± .01	20.1 ± 1.1	.06 ± .03	22.4 ± 0.4
	$sf = 8$	.06 ± .03	21.4 ± 0.5	.05 ± .02	18.8 ± 1.0	.03 ± .02	18.4 ± 1.0
	$sf = 16$	.03 ± .03	20.0 ± 0.8	.05 ± .01	18.3 ± 0.5	.03 ± .01	18.3 ± 0.9

Table 1: Rank correlation coefficient and saliency pixel intersection on the test set, for the interpretations of neural nets trained with a training set split factor of $\mathrm{sf} = 2, 4, 8, 16$.

centered perturbations with the attacker network for the source network. The intuition behind this technique is that if the networks have similar gradient interpretations, then the perturbations generated by the attacker would have negligible effect on the source networks' saliency map outputs. In Table 1, we compare the *a) rank correlation of saliency maps*, the Spearman rank correlation coefficient between the saliency maps of the original and perturbed images; *b) top-100 salient pixel intersection %*, indicating the percentage of overlap between the top-100 most salient pixels, which are used for classifying the original and perturbed images. Our comparison shows a consistent improvement of the discussed metrics by increasing the training set size.

6.4. Improving Generalization via SmoothGrad

We conduct experiments with the simple gradient [1] and the SmoothGrad [32] methods. Our goals are to first quantify the within-model discrepancy (misattribution) between the input and output and second to evaluate how the cross-network gradient-based interpretations increasingly disagree with fewer training samples. We subsequently compute the mean L_2-norm difference of the interpretation vectors for networks with disjoint training sets of the same size. A larger norm difference indicates a greater discrepancy between the interpretations and worse generalization.

We report results averaging over m = 1, 5, 20, 50 Gaussian noise vectors for the estimation of the SmoothGrad interpretation, with Gaussian perturbation standard deviation σ chosen from the set $\{0, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0\}$. In the experiments, we observe that increasing the number of randomly-perturbed samples with Gaussian noise has a gradient smoothing effect. Also, as visualized in Fig. 4, increasing the noise standard deviation improves Gaussian smoothing power, with effects of increasing the interpretation agreement and reducing the generalization gap. Comparing the simple gradient (marked by "no σ" in the legends) and SmoothGrad methods' results, we observe that the Gaussian smoothing in SmoothGrad improves cross-network interpretation agreement and hence the generalization of the gradient-based saliency map. This observation is consistent with our theoretical analysis, evidencing the regularization role of Gaussian smoothing in SmoothGrad.

7. Conclusion

In this paper, we highlighted the role of proper generalization from training samples to unseen test data in the success of deep learning-based interpretation methods. On the theory side, we proved generalization error bounds to show the higher sample complexity of learning interpretable neural net classifiers, and further discussed the regularization effect of Gaussian smoothing in the SmoothGrad approach. On the empirical side, our numerical results also demonstrate the influence of the training set size on the generalization of gradient-based interpretation methods to test samples. To further expand the analysis, an interesting future direction is to explore other regularization schemes and their effect on the generalization of interpretation methods. Such a study can be performed for popular deep learning regularization schemes such as batch normalization and dropout. Furthermore, the extensions of our generalization study to mask-based and perturbation-based explanation tools could improve the understanding of the effect of adversarial schemes on the generalization properties of the interpretability of neural networks. We note that our developed generalization framework is relatively general and potentially applicable for studying the discussed future directions.

References

[1] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. How to explain individual classification decisions. *Journal of Machine Learning Research*, 11(61):1803–1831, 2010.

[2] Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. *Advances in neural information processing systems*, 30, 2017.

[3] Lucas Beyer, Olivier J Hénaff, Alexander Kolesnikov, Xiaohua Zhai, and Aäron van den Oord. Are we done with imagenet? *arXiv preprint arXiv:2006.07159*, 2020.

[4] Lucas Beyer, Xiaohua Zhai, and Alexander Kolesnikov. Better plain vit baselines for imagenet-1k. *arXiv preprint arXiv:2205.01580*, 2022.

[5] Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek. Layer-wise relevance propagation for neural networks with local renormalization layers. In *International Conference on Artificial Neural Networks*, pages 63–71. Springer, 2016.

[6] Ann-Kathrin Dombrowski, Maximillian Alber, Christopher Anders, Marcel Ackermann, Klaus-Robert Müller, and Pan Kessel. Explanations can be manipulated and geometry is to blame. *Advances in Neural Information Processing Systems*, 32, 2019.

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[8] Thomas Fel, David Vigouroux, Rémi Cadène, and Thomas Serre. How good is your explanation? algorithmic stability measures to assess the quality of explanations for deep neural networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 720–730, January 2022.

[9] Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile. *AAAI*, 33(01):3681–3688, Jul. 2019.

[10] Griffin, Holub, and Perona. Caltech 256, Apr 2022.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[12] Juyeon Heo, Sunghwan Joo, and Taesup Moon. Fooling neural network interpretations via adversarial model manipulation. *Advances in Neural Information Processing Systems*, 32, 2019.

[13] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International Conference on Machine Learning*, pages 2668–2677. PMLR, 2018.

[14] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In Hal Daumé III and Aarti Singh, editors, *International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5338–5348. PMLR, 13–18 Jul 2020.

[15] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pages 3519–3529. PMLR, 2019.

[16] Alex Krizhevsky et al. Learning multiple layers of features from tiny images. 2009.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[18] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. 2015.

[19] Alexander Levine, Sahil Singla, and Soheil Feizi. Certifiably robust interpretation in deep learning. *arXiv preprint arXiv:1905.12105*, 2019.

[20] Zachary C Lipton. The mythos of model interpretability. *arXiv preprint arXiv:1606.03490*, 2016.

[21] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88, 2017.

[22] Jinlong Liu, Yunzhi Bai, Guoqing Jiang, Ting Chen, and Huayan Wang. Understanding why neural networks generalize well through gsnr of parameters. In *ICLR*, 2020.

[23] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *CVPR*, pages 11976–11986, 2022.

[24] Ari Morcos, Maithra Raghu, and Samy Bengio. Insights on representational similarity in neural networks with canonical correlation. *Advances in Neural Information Processing Systems*, 31, 2018.

[25] Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. *Advances in neural information processing systems*, 30, 2017.

[26] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *NIPS*, 2021.

[27] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015.

[28] 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 *CVPR*, pages 618–626, 2017.

[29] Vaishaal Shankar, Rebecca Roelofs, Horia Mania, Alex Fang, Benjamin Recht, and Ludwig Schmidt. Evaluating machine accuracy on ImageNet. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 8634–8644. PMLR, 13–18 Jul 2020.

[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, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

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

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

[34] Pierre Stock and Moustapha Cisse. Convnets and imagenet beyond accuracy: Understanding mistakes and uncovering biases. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 498–512, 2018.

[35] Akshayvarun Subramanya, Vipin Pillai, and Hamed Pirsiavash. Fooling network interpretation in image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2020–2029, 2019.

[36] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR, 2017.

[37] Mingxing Tan and Quoc Le. Efficientnetv2: Smaller models and faster training. In *International Conference on Machine Learning*, pages 10096–10106. PMLR, 2021.

[38] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *NIPS*, 34:24261–24272, 2021.

[39] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pages 818–833. Springer, 2014.

[40] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *ICLR*, 2017.

[41] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu. Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, 30(11):3212–3232, 2019.

[42] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, pages 2921–2929, 2016.

[43] Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. Interpretable basis decomposition for visual explanation. In *ECCV*, September 2018.

Generalization of Interpretable Deep Learning Requires More Data (Supplementary Material)

1. Proofs

1.1. Proof of Theorem 1

We begin by proving the following lemmas.

Lemma 1. *Under Theorem 1's assumptions, the neural network score function's gradient $\nabla_{\mathbf{x}} f_{\mathbf{w},c}$ satisfies the following perturbation error bound when the weight matrix W_k is perturbed by a norm-bounded matrix Δ_k for which $\|\Delta_k\|_2 \le t$, where we define $\widetilde{\mathbf{w}} = \mathrm{vec}(W_1,\dots,W_{k-1},W_k+\Delta_k,W_{k+1},\dots,W_L)$,*

$$\begin{aligned}&\big\|\nabla_{\mathbf{x}} f_{\mathbf{w},c}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\tilde{\mathbf{w}},c}(\mathbf{x})\big\| \\ &\le \frac{L_{\mathbf{w}}\sum_{i=k}^{L}\prod_{j=1}^{i}\gamma_j\|W_j\|}{\|W_k\|_2}\|\Delta_k\|_2\end{aligned}$$

Proof. The neural net's gradient with respect to input $\mathbf{x}$ is as follows:

$$\nabla_{\mathbf{x}} f_{\mathbf{w},c}(\mathbf{x}) = \mathbf{w}_{L,c}^{\top}\prod_{i=1}^{L-1} W_i^{\top}\,\mathrm{diag}\big(\phi_i'(h_{\mathbf{w_{1:i}}}(\mathbf{x}))\big).$$

Here, $f_{\mathbf{w_{1:i}}}(\mathbf{x})$ is the neural net's output at layer i. As a result, since $\widetilde{\mathbf{w}}$ is different from $\mathbf{w}$ only at layer k we obtain:

$$\begin{aligned}&\big\|\nabla_{\mathbf{x}} f_{\mathbf{w},c}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\tilde{\mathbf{w}},c}(\mathbf{x})\big\|_2 \\ &\le \sum_{i=k}^{L}\Big[\big(\prod_{j=1}^{L}\gamma_j\|W_j\|_2\big)\big(\prod_{j=1}^{i}\gamma_j\|W_j\|_2\big)\Big]\frac{\|\Delta_k\|_2}{\|W_k\|_2} \\ &= \Big(\prod_{j=1}^{L}\gamma_j\|W_j\|_2\Big)\sum_{i=k}^{L}\Big[\prod_{j=1}^{i}\gamma_j\|W_j\|_2\Big]\frac{\|\Delta_k\|_2}{\|W_k\|_2} \\ &= \frac{L_{\mathbf{w}}\sum_{i=k}^{L}\prod_{j=1}^{i}\gamma_j\|W_j\|}{\|W_k\|_2}\|\Delta_k\|_2.\end{aligned}$$

The proof is therefore complete. □

Lemma 2. *Under Theorem 1's assumptions, the neural network's integrated gradients Int-Grad$(f_{\mathbf{w},c},\mathbf{x})$ satisfies the following perturbations error bound when the weight matrix W_k is perturbed by a norm-bounded matrix Δ_k such that $\|\Delta_k\|_2 \le t$, where we define $\widetilde{\mathbf{w}} = \mathrm{vec}(W_1,\dots,W_{k-1},W_k+\Delta_k,W_{k+1},\dots,W_L)$:*

$$\begin{aligned}&\big\|\textit{Int-Grad}(f_{\mathbf{w},c},\mathbf{x}) - \textit{Int-Grad}(f_{\tilde{\mathbf{w}},c},\mathbf{x})\big\| \\ &\le \frac{\|\mathbf{x}-\mathbf{x}^0\|_\infty L_{\mathbf{w}}\sum_{i=k}^{L}\prod_{j=1}^{i}\gamma_j\|W_j\|}{\|W_k\|_2}\|\Delta_k\|_2\end{aligned}$$

Proof. Note that according to the definition, we have:

$$\text{Int-Grad}(\mathbf{f}_c,\mathbf{x}) := \int_0^1 \nabla_{\mathbf{x}}\mathbf{f}_c\big(\mathbf{x}^0+\alpha\Delta\mathbf{x}\big)\odot\Delta\mathbf{x}\,\mathrm{d}\alpha$$

As shown in Lemma 1, the weight perturbation will lead to a bounded change to the gradient at every $\mathbf{x}'$

$$\begin{aligned}&\big\|\nabla_{\mathbf{x}} f_{\mathbf{w},c}(\mathbf{x}') - \nabla_{\mathbf{x}} f_{\tilde{\mathbf{w}},c}(\mathbf{x}')\big\| \\ &\le \frac{L_{\mathbf{w}}\sum_{i=k}^{L}\prod_{j=1}^{i}\gamma_j\|W_j\|}{\|W_k\|_2}\|\Delta_k\|_2.\end{aligned}$$

Therefore, we will have:

$$\begin{aligned}&\big\|\nabla_{\mathbf{x}} f_{\mathbf{w},c}(\mathbf{x}')\odot(\mathbf{x}-\mathbf{x}^0) - \nabla_{\mathbf{x}} f_{\tilde{\mathbf{w}},c}(\mathbf{x}')\odot(\mathbf{x}-\mathbf{x}^0)\big\| \\ &= \big\|\big(\nabla_{\mathbf{x}} f_{\mathbf{w},c}(\mathbf{x}') - \nabla_{\mathbf{x}} f_{\tilde{\mathbf{w}},c}(\mathbf{x}')\big)\odot(\mathbf{x}-\mathbf{x}^0)\big\| \\ &\le \frac{\|\mathbf{x}-\mathbf{x}^0\|_\infty L_{\mathbf{w}}\sum_{i=k}^{L}\prod_{j=1}^{i}\gamma_j\|W_j\|}{\|W_k\|_2}\|\Delta_k\|_2\end{aligned}$$

where the last line follows the inequality $\|\mathbf{a}\odot\mathbf{b}\|_2 \le \|\mathbf{a}\|_2\|\mathbf{b}\|_\infty$. The lemma's proof is thus complete. □

To prove Theorem 1 in the case of the simple gradient approach, we use a similar covering-number-based approach to [1]'s analysis for standard classification deep learning problems. Therefore, we suppose the norm constraints $\|W_i\|_2 \le a_i$, $\|W_i\|_{2,1} \le b_i$ for each $i = 1,\dots,L$. Next, we use the following covering resolution parameters:

$$\epsilon_k = \frac{a_k\alpha_k\epsilon}{\big(\prod_{i=1}^{L}\gamma_i a_i\big)\big(\sum_{i=k}^{L}\prod_{j=1}^{i}\gamma_j a_j\big)},$$

$$\text{where}\quad \alpha_k = \frac{1}{A}\frac{{b_k}^{2/3}}{{a_k}^{2/3}},\ A = \sum_{i=1}^{L}\frac{{b_i}^{2/3}}{{a_i}^{2/3}}$$

Note that Lemma 1 implies that by finding an ϵ_k-covering for each W_k, the covering resolution for $[\nabla_{\mathbf{x}} f_{\mathbf{w}}(\mathbf{x})]|_S$ (S is the training set) will be upper-bounded by

$$\sum_{k=1}^{L}\Big[\frac{L_{\mathbf{w}}\sum_{i=k}^{L}\prod_{j=1}^{i}\gamma_j\|W_j\|}{\|W_k\|_2}\epsilon_k\Big]=\epsilon.$$

Hence, using Lemma A.7 from [1] will result in the following bound on the ϵ-covering-number for the set $[\nabla_{\mathbf{x}}\mathcal{F}_{\mathcal{W}}-\nabla_{\mathbf{x}}f^*]|_S=\{\nabla_{\mathbf{x}}f_{\mathbf{w},c}(\mathbf{X})-\nabla_{\mathbf{x}}f^*_{c(X)}(\mathbf{X}):\ \forall 1\le i\le L:\|W_i\|_2\le a_i,\ \|W_i\|_{2,1}\le b_i\}$

$$\begin{aligned}
&\log\mathcal{N}\big([\nabla_{\mathbf{x}}\mathcal{F}_{\mathcal{W}}-\nabla_{\mathbf{x}}f^*]|_S,\|\cdot\|_2,\epsilon\big)\\
&\le\sum_{i=1}^{L}\sup_{\mathbf{w}_{-i}\in\mathcal{W}}\Big[\log\mathcal{N}\big(\nabla(f_{\mathbf{w},c(\mathbf{X})}-f^*_{c(\mathbf{X})})(\mathbf{X}):\\
&\qquad\|\mathbf{W}_i\|_2\le a_i,\ \|\mathbf{W}_i\|_{2,1}\le b_i\big\},\|\cdot\|_2,\epsilon_i\big)\Big]\\
&\le\sum_{i=1}^{L}\sup_{\mathbf{w}_{-i}\in\mathcal{W}}\Big[\log\mathcal{N}\big(\nabla(f_{\mathbf{w},c(\mathbf{X})}-f^*_{c(\mathbf{X})})(\mathbf{X}):\\
&\qquad\|\mathbf{W}_i\|_{2,1}\le b_i\big\},\|\cdot\|_2,\epsilon_i\big)\Big]\\
&\le\sum_{i=1}^{L}\Big[\sup_{\mathbf{w}_{-i}\in\mathcal{W}}\frac{{b_i}^2\|\nabla(f_{\mathbf{w},c(\mathbf{X})}-f^*_{c(\mathbf{X})})(\mathbf{X})\|_2^2}{{\epsilon_i}^2}\log(2W^2)\Big]\\
&\le\sum_{i=1}^{L}\big[\sup_{\mathbf{w}_{-i}\in\mathcal{W}}\frac{L_{\mathbf{w}}^2B^2\log(2W^2)}{\epsilon^2}\frac{{b_i}^2}{{\epsilon_i}^2}\big]\\
&\le\frac{\log(2W^2)B^2\prod_{i=1}^{L}\gamma_i^2{a_i}^2}{\epsilon^2}\\
&\qquad\times\sum_{k=1}^{L}\big[\frac{{b_k}^2(\sum_{i=k}^{L}\prod_{j=1}^{i}\gamma_ja_j)^2}{{\alpha_k}^2a_k^2}\big]\\
&\le\frac{\log(2W^2)B^2\prod_{i=0}^{L}\gamma_i^2{a_i}^2(\sum_{i=1}^{L}\prod_{j=1}^{i}\gamma_ia_i)^2}{\epsilon^2}\\
&\qquad\times\sum_{k=0}^{L}\big[\frac{{b_k}^2}{{\alpha_k}^2a_k^2}\big]\\
&\le\frac{4\log(2W^2)B^2\prod_{i=1}^{L}\gamma_i^2{a_i}^2\big(\sum_{i=1}^{L}\prod_{j=1}^{i}\gamma_ja_j\big)^2}{\epsilon^2}\\
&\qquad\times\sum_{i=1}^{L}\big[\frac{{b_i}^2}{{\alpha_i}^2a1_i^2}\big]\\
&=\frac{C}{\epsilon^2}
\end{aligned}$$

where C is defined as

$$C=4\log(2W^2)B^2\Bigg[\prod_{i=1}^{L}\gamma_i^2{a_i}^2\big(\sum_{i=1}^{L}\prod_{j=1}^{i}\gamma_ja_j\big)^2\Bigg[\sum_{i=1}^{L}\frac{{b_i}^{2/3}}{{a_i}^{2/3}}\Bigg]^3\Bigg]$$

Using the above covering-number bound, we apply the Dudley entropy integral bound [1] which bounds the Rademacher complexity of $[\|\nabla\mathcal{F}-\nabla f^*\|_2]|_S$ as

$$\begin{aligned}
&\mathcal{R}([\|\nabla\mathcal{F}-\nabla f^*\|_2]|_S)\\
&\le\inf_{\alpha\ge0}\Big\{\frac{4\alpha}{\sqrt{n}}\\
&+\frac{12}{n}\int_{\alpha}^{\sqrt{n}}\sqrt{\log\mathcal{N}\big(\|\nabla\mathcal{F}(\mathbf{X})-\nabla f^*(\mathbf{X})\|_2,\epsilon\big)}d\epsilon\Big\}\\
&\le\inf_{\alpha\ge0}\Big\{\frac{4\alpha}{\sqrt{n}}+\frac{12\sqrt{C}}{n}\log(\frac{\sqrt{n}}{\alpha})\Big\}\\
&\le\frac{4}{n^{3/2}}+\frac{18\log(n)\sqrt{C}}{n}\\
&\le\frac{4}{n^{3/2}}+\Bigg[\frac{18\log(n)}{n}\times\\
&\quad 4\log(2W^2)B\Bigg[\prod_{i=1}^{L}\gamma_ia_i\big(\sum_{i=1}^{L}\prod_{j=1}^{i}\gamma_ja_j\big)\Bigg[\sum_{i=1}^{L}\frac{{b_i}^{2/3}}{{a_i}^{2/3}}\Bigg]^{3/2}\Bigg]
\end{aligned}$$

Here, the last two inequalities come from choosing $\alpha=1/n$. Therefore, we have the following bound where $R_{\mathcal{W}}=\sup_{\mathbf{w}\in\mathcal{W}}R_{\mathbf{w}}$ as defined in Theorem 1:

$$\begin{aligned}
&\mathcal{R}([\|\nabla\mathcal{F}-\nabla f^*\|_2]|_S)\\
&\le\mathcal{O}\Big(\frac{BL_{\mathbf{w}}R_{\mathcal{W}}\log(n)\log(D)}{n}\big(\sum_{i=1}^{L}\frac{\|W_i\|_{2,1}^{2/3}}{\|W_i\|_2^{2/3}}\big)^{3/2}\Big)
\end{aligned}$$

Therefore, according to the well-known Rademacher complexity-based generalization analysis [?], for every $\omega>0$ with probability at least $1-\omega$ we have for every $\mathbf{w}\in\mathcal{W}$:

$$\begin{aligned}
&\frac{1}{n}\sum_{i=1}^{n}[\|\mathrm{SG}(f_{\mathbf{w},c},\mathbf{x}_i)-\mathrm{SG}(f_c^*,\mathbf{x}_i)\|_2]\\
&-\mathbb{E}_{X\sim P_{\mathbf{X}}}[\|\mathrm{SG}(f_{\mathbf{w},c},X)-\mathrm{SG}(f_c^*,X)\|_2]\\
&\le\mathcal{O}\Big(c\sqrt{\frac{\log(1/\omega)}{n}}\\
&\quad+\frac{BL_{\mathbf{w}}R_{\mathcal{W}}\log(n)\log(D)}{n}\big(\sum_{i=1}^{L}\frac{\|W_i\|_{2,1}^{2/3}}{\|W_i\|_2^{2/3}}\big)^{3/2}\Big),
\end{aligned}$$

which is the same as

$$\begin{aligned}
\epsilon_{\text{gen}}(f_{\mathbf{w}})\ \le\ &\mathcal{O}\Big(c\sqrt{\frac{\log(1/\omega)}{n}}\\
&+\frac{BL_{\mathbf{w}}R_{\mathcal{W}}\log(n)\log(D)}{n}\big(\sum_{i=1}^{L}\frac{\|W_i\|_{2,1}^{2/3}}{\|W_i\|_2^{2/3}}\big)^{3/2}\Big).
\end{aligned}$$

Therefore the proof is complete for the SimpleGrad interpretation.

In order to prove Theorem 1 for the Integrated Gradients case, we can follow the same steps we used for the SimpleGrad case with the difference that as shown in Lemma 2 the perturbation bound for the Integrated Gradients case is by a multiplicative factor $\|\mathbf{x}-\mathbf{x}^0\|_\infty$ larger than the case of simple gradients. In addition, the supremum of the ℓ_2-operator-norm difference of $\|\text{Int-Grad}(f_{\mathbf{w},c},\mathbf{X}) - \text{Int-Grad}(f^*_c,\mathbf{X})\|_2$ is also by the same multiplicative factor $\|\mathbf{x}-\mathbf{x}^0\|_\infty$ larger than the case of simple gradients. Therefore, assuming that $\|\mathbf{x}-\mathbf{x}^0\|_\infty \le E$ holds with probability 1 for constant E, we can follow the same steps of the proof for the simple gradient to show the following for the integrated Gradients:

$$\epsilon_{\text{gen}}(f_{\mathbf{w}}) \le \mathcal{O}\biggl(c\sqrt{\frac{\log(1/\omega)}{n}} + \frac{BL_{\mathbf{w}}R_{\mathcal{W}}E^2\log(n)\log(D)}{n}\bigl(\sum_{i=1}^L \frac{\|W_i\|_{2,1}^{2/3}}{\|W_i\|_2^{2/3}}\bigr)^{3/2}\biggr).$$

The proof is therefore complete.

1.1.1 Proof of Theorem 2

We begin by proving the following Lemma.

Lemma 3. *Under Theorem 2's assumptions, the neural network's smooth gradient SmoothGrad$(f_{\mathbf{w},c},\mathbf{x})$ satisfies the following perturbation error bound when the weight matrix W_k is perturbed by a norm-bounded matrix Δ_k such that $\|\Delta_k\|_2 \le t$, where we define $\widetilde{\mathbf{w}} = \text{vec}(W_1,\dots,W_{k-1},W_k+\Delta_k,W_{k+1},\dots,W_L)$:*

$$\begin{aligned}&\bigl\|\mathit{SmoothGrad}(f_{\mathbf{w},c},\mathbf{x}) - \mathit{SmoothGrad}(f_{\mathbf{w},c},\mathbf{x})\bigr\|_2\\ &\le \mathbb{O}\left(\max\{\frac{B\sqrt{d}}{\sigma},d\}\frac{L_{\mathbf{w}}}{\sigma\|W_k\|_2}\|\Delta_k\|_2\right)\end{aligned}$$

Proof. To prove this result, we apply Stein's lemma (Lemma 1 in the main text) which shows that

$$\text{SmoothGrad}(f_{\mathbf{w},c},\mathbf{x}) = \mathbb{E}_{Z\sim\mathcal{N}(0,\sigma^2 I)}\bigl[\frac{Z}{\sigma^2}f_{\mathbf{w},c}(\mathbf{x}+Z)\bigr].$$

Therefore, we have:

$$\begin{aligned}&\bigl\|\text{SmoothGrad}(f_{\mathbf{w},c},\mathbf{X}) - \text{SmoothGrad}(f_{\mathbf{w},c},\mathbf{X})\bigr\|_2\\ =:&\left\|\mathbb{E}\bigl[\frac{\mathbf{Z}}{\sigma^2}(f_{\mathbf{w},c},\mathbf{X}+\mathbf{Z}) - (f_{\mathbf{w},c},\mathbf{X}+\mathbf{Z})\bigr]\right\|_2\\ \le:&\mathbb{E}\left[\bigl\|\frac{\mathbf{Z}}{\sigma^2}\bigr\|_2\bigl|f_{\mathbf{w},c},\mathbf{X}+\mathbf{Z}) - (f_{\mathbf{w},c},\mathbf{X}+\mathbf{Z})\bigr|\right]\\ \le&\,\mathbb{E}\left[\frac{L_{\mathbf{w}}\|\mathbf{X}+\mathbf{Z}\|_2}{\|W_k\|_2}\|\Delta_k\|_2\frac{\|\mathbf{Z}\|_2}{\sigma^2}\right]\\ \le&\,\frac{L_{\mathbf{w}}\|\Delta_k\|_2}{\|W_k\|_2}\mathbb{E}\bigl[(\|\mathbf{X}\|_2+\|\mathbf{Z}\|_2)\frac{\|\mathbf{Z}\|_2}{\sigma^2}\bigr]\\ \le&\,\frac{L_{\mathbf{w}}\|\Delta_k\|_2}{\|W_k\|_2}\mathbb{O}\left(\max\{\frac{B\sqrt{d}}{\sigma},d\}\right)\end{aligned}$$

The last inequality follows from the well-known facts [9] that $\mathbb{E}[\|\mathbf{Z}\|_2] \le c\sigma\sqrt{d}$ holds for a universal constant c and a random matrix distribution with independent Gaussian entries distributed as $\mathcal{N}(0,\sigma^2)$, and also $\mathbb{E}[\|\mathbf{Z}\|_2^2] \le d\sigma^2$ as $\|\mathbf{Z}\|_2^2$ is the maximum eigenvalue of both $\mathbf{Z}^\top\mathbf{Z}$ and $\mathbf{Z}\mathbf{Z}^\top$. Hence, the lemma's proof is complete. ◻

To prove Theorem 2, we again apply a similar covering-number-based approach to [1]'s analysis. Therefore, we suppose the norm constraints $\|W_i\|_2 \le a_i$, $\|W_i\|_{2,1} \le b_i$ for each $i = 1,\dots,L$. Next, we use the following covering resolution parameters:

$$\begin{aligned}\epsilon_k &= \frac{a_k\alpha_k\epsilon}{(\prod_{i=1}^k\gamma_i a_i)c\max\{\frac{B\sqrt{d}}{\sigma},d\}},\\ \text{where}\quad \alpha_k &= \frac{1}{A}\frac{b_k{}^{2/3}}{a_k{}^{2/3}},\ A = \sum_{i=1}^L\frac{b_i{}^{2/3}}{a_i{}^{2/3}}\end{aligned}$$

In the above c is the constant that is required for the order-wise bound of Lemma 3 to hold with equality. Note that Lemma 3 shows that if we have an ϵ_k-covering for each W_k, the covering resolution for $\|\text{SmoothGrad}(f_{\mathbf{w}},\mathbf{X}) - \text{SmoothGrad}(f^*,\mathbf{X})\|_2|_S$ (S is the training set) will be upper-bounded by

$$\sum_{k=1}^L\left[\frac{c(\prod_{i=1}^k\gamma_i a_i)\max\{\frac{B\sqrt{d}}{\sigma},d\}}{\|W_k\|_2}\epsilon_k\right] = \epsilon.$$

Hence, using Lemma A.7 from [1] will result in the following bound on the ϵ-covering-number for the set $\|\text{SmoothGrad}(f_{\mathbf{w}},\mathbf{X}) - \text{SmoothGrad}(f^*,\mathbf{X})\|_2|_S$: $\forall 1\le i\le L : \|W_i\|_2 \le a_i,\ \|W_i\|_{2,1} \le b_i\}$

$$\begin{aligned}&\log\mathcal{N}\bigl(\|\text{SmoothGrad}(f_{\mathbf{w}},\mathbf{X})\\ &\quad - \text{SmoothGrad}(f^*,\mathbf{X})\|_2|_S,\|\cdot\|_2,\epsilon\bigr)\\ \le&\sum_{i=1}^L\sup_{\mathbf{w}_{-i}\in\mathcal{W}}\Bigl[\log\mathcal{N}\bigl(\text{SmoothGrad}(f_{\mathbf{w},c(\mathbf{X})} - f^*_{c(\mathbf{X})})(\mathbf{X}) :\\ &\quad \|\mathbf{W}_i\|_2 \le a_i,\ \|\mathbf{W}_i\|_{2,1} \le b_i\},\|\cdot\|_2,\epsilon_i\bigr)\Bigr]\\ \le&\sum_{i=1}^L\sup_{\mathbf{w}_{-i}\in\mathcal{W}}\Bigl[\log\mathcal{N}\bigl(\text{SmoothGrad}(f_{\mathbf{w},c(\mathbf{X})} - f^*_{c(\mathbf{X})})(\mathbf{X}) :\\ &\quad \|\mathbf{W}_i\|_{2,1} \le b_i\},\|\cdot\|_2,\epsilon_i\bigr)\Bigr]\\ \le&\sum_{i=1}^L\Bigl[\sup_{\mathbf{w}_{-i}\in\mathcal{W}}\end{aligned}$$

$$\frac{{b_i}^2 \|\text{SmoothGrad}(f_{\mathbf{w},c(\mathbf{X})} - f^*_{c(\mathbf{X})})(\mathbf{X})\|_2^2}{{\epsilon_i}^2} \log(2W^2)\Big]$$

$$\leq \sum_{i=1}^{L} \Big[\sup_{\mathbf{w}_{-i} \in \mathcal{W}} \frac{L_{\mathbf{w}}^2 (B^2 + \sigma^2 d) \log(2W^2)}{\epsilon^2} \frac{{b_i}^2}{{\epsilon_i}^2} \Big]$$

$$\leq \frac{\log(2W^2)(B^2 + \sigma^2 d) \prod_{i=1}^{L} \gamma_i^2 {a_i}^2}{\epsilon^2}$$

$$\times \sum_{k=1}^{L} \Big[\frac{{b_k}^2 (\sum_{i=k}^{L} \prod_{j=1}^{i} \gamma_j a_j)^2}{{\alpha_k}^2 a_k^2} \Big]$$

$$\leq \frac{\log(2W^2)(B^2 + \sigma^2 d) \prod_{i=0}^{L} \gamma_i^2 {a_i}^2 (\sum_{i=1}^{L} \prod_{j=1}^{i} \gamma_i a_i)^2}{\epsilon^2}$$

$$\times \sum_{k=0}^{L} \Big[\frac{{b_k}^2}{{\alpha_k}^2 a_k^2} \Big]$$

$$\leq \frac{4 \log(2W^2)(B^2 + \sigma^2 d) \prod_{i=1}^{L} \gamma_i^2 {a_i}^2 \big(\sum_{i=1}^{L} \prod_{j=1}^{i} \gamma_j a_j\big)^2}{\epsilon^2}$$

$$\times \sum_{i=1}^{L} \Big[\frac{{b_i}^2}{{\alpha_i}^2 a 1_i^2} \Big]$$

$$= \frac{C}{\epsilon^2}$$

where C is

$$C = 4 \log(2W^2)(B^2 + \sigma^2 d) \times \left[\prod_{i=1}^{L} \gamma_i^2 {a_i}^2 \Big(\sum_{i=1}^{L} \prod_{j=1}^{i} \gamma_j a_j\Big)^2 \left[\sum_{i=1}^{L} \frac{{b_i}^{2/3}}{{a_i}^{2/3}} \right] \right]^3$$

With the above covering-number bound, we use the Dudley entropy integral bound [1] bounding the Rademacher complexity of $\|\text{SmoothGrad}(f_{\mathbf{w}}, \mathbf{X}) - \text{SmoothGrad}(f^*, \mathbf{X})\|_2|_S$ as

$$\mathcal{R}(\|\text{SmoothGrad}(f_{\mathbf{w}}, \mathbf{X}) - \text{SmoothGrad}(f^*, \mathbf{X})\|_2|_S)$$

$$\leq \inf_{\alpha \geq 0} \left\{ \frac{4\alpha}{\sqrt{n}} \right.$$

$$\leq \inf_{\alpha \geq 0} \left\{ \frac{4\alpha}{\sqrt{n}} + \frac{12\sqrt{C}}{n} \log(\frac{\sqrt{n}}{\alpha}) \right\}$$

$$\leq \frac{4}{n^{3/2}} + \frac{18 \log(n) \sqrt{C}}{n}$$

$$\leq \frac{4}{n^{3/2}} + \Bigg[\frac{18 \log(n)}{n} 4 \log(2W^2)(B + \sigma\sqrt{d}) \times \left[\prod_{i=1}^{L} \gamma_i a_i \Big(\sum_{i=1}^{L} \prod_{j=1}^{i} \gamma_j a_j\Big) \left[\sum_{i=1}^{L} \frac{{b_i}^{2/3}}{{a_i}^{2/3}} \right] \right]^{3/2} \Bigg]$$

Here, the last two inequalities follow from choosing $\alpha = 1/n$. Therefore, we have the following bound:

$$\mathcal{R}(\|\text{SmoothGrad}(f_{\mathbf{w}}, \mathbf{X}) - \text{SmoothGrad}(f^*, \mathbf{X})\|_2|_S)$$

$$\leq \mathcal{O}\Bigg(\frac{(B + \sigma\sqrt{d}) L_{\mathbf{w}} \log(n) \log(D)}{n} \Big(\sum_{i=1}^{L} \frac{\|W_i\|_{2,1}^{2/3}}{\|W_i\|_2^{2/3}} \Big)^{3/2} \Bigg)$$

Therefore, based on the Rademacher complexity-based generalization bound [?], for every $\omega > 0$ with probability at least $1 - \omega$ we have for every $\mathbf{w} \in \mathcal{W}$:

$$\frac{1}{n} \sum_{i=1}^{n} [\|\text{SmoothGrad}(f_{\mathbf{w},c}, \mathbf{x}_i) - \text{SmoothGrad}(f_c^*, \mathbf{x}_i)\|_2]$$

$$- \mathbb{E} [\|\text{SmoothGrad}(f_{\mathbf{w},c}, X) - \text{SmoothGrad}(f_c^*, X)\|_2]$$

$$\leq \mathcal{O}\Bigg(c\sqrt{\frac{\log(1/\omega)}{n}} + \frac{\sqrt{d}(B + \sqrt{d}\sigma) L_{\mathbf{w}} \log(n) \log(D)}{\sigma n} \Big(\sum_{i=1}^{L} \frac{\|W_i\|_{2,1}^{2/3}}{\|W_i\|_2^{2/3}} \Big)^{3/2} \Bigg),$$

which assuming that $B = \Omega(\sqrt{d}\sigma)$ is the same as

$$\epsilon_{\text{gen}}(f_{\mathbf{w}}) \leq \mathcal{O}\Bigg(c\sqrt{\frac{\log(1/\omega)}{n}} + \frac{\sqrt{d} B L_{\mathbf{w}} \log(n) \log(D)}{\sigma n} \Big(\sum_{i=1}^{L} \frac{\|W_i\|_{2,1}^{2/3}}{\|W_i\|_2^{2/3}} \Big)^{3/2} \Bigg).$$

Thus, the proof is complete.

2. Experimental Results

2.1. On the Generalization Gap

Visualizing Feature Importance. We verify the observation that generalizable interpretations require more data through qualitative gradient-based methods. Figure 3 contains a full version of qualitative experiment results on Caltech-256 [2], with the simple gradient method [6], SmoothGrad [7], integrated gradients [8] and DeepLift [5]. We observe that large-scale pre-training on ImageNet significantly improves model interpretations, in terms of visual fidelity, localization meaningfulness and generalization ability to unseen test samples. Figures 1 and 2 depict results for convolutional neural network architectures, where we vary the number of training samples from Pretrain, to 50%, 25%, 12.5%, 6.25%, then visualize the localization intersections between GradCAM [4] interpretations. We consistently observe that while it is possible for models trained on 6.25% of the training set to generate aligned activation maps on seen training samples, they are unable to transfer this localization agreement to unseen test samples.

Quantifying Feature Differences. Using integrated gradients [8] in Table 1, we report the rank correlation and top-100 salient pixels intersection. Most importantly, as the split factor increases, the generalization

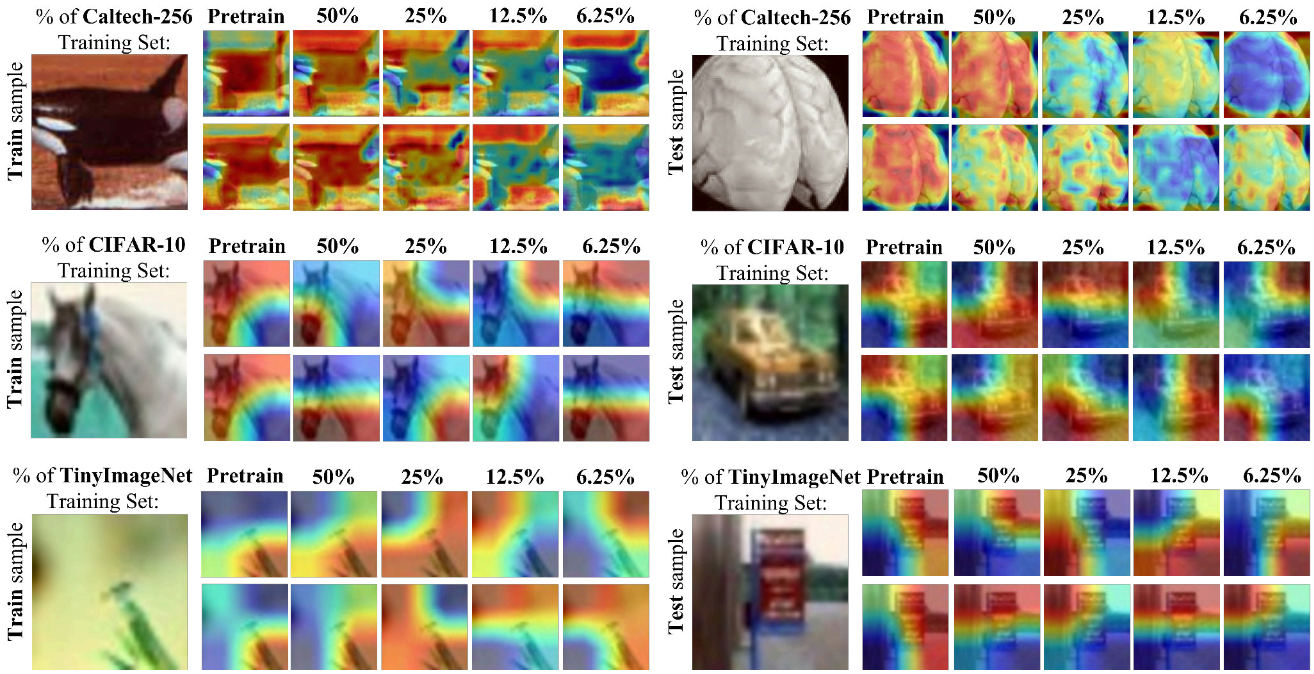


Figure 1: Grad-CAM comparisons with EfficientNet-V2-S. As we increase the number of training samples from 6.25% ($sf = 16$) of the training set, to using 50% of the training set, then to pre-training on ImageNet plus fine-tuning with 50% training data, we observe that model pairs generate increasingly consistent interpretations.

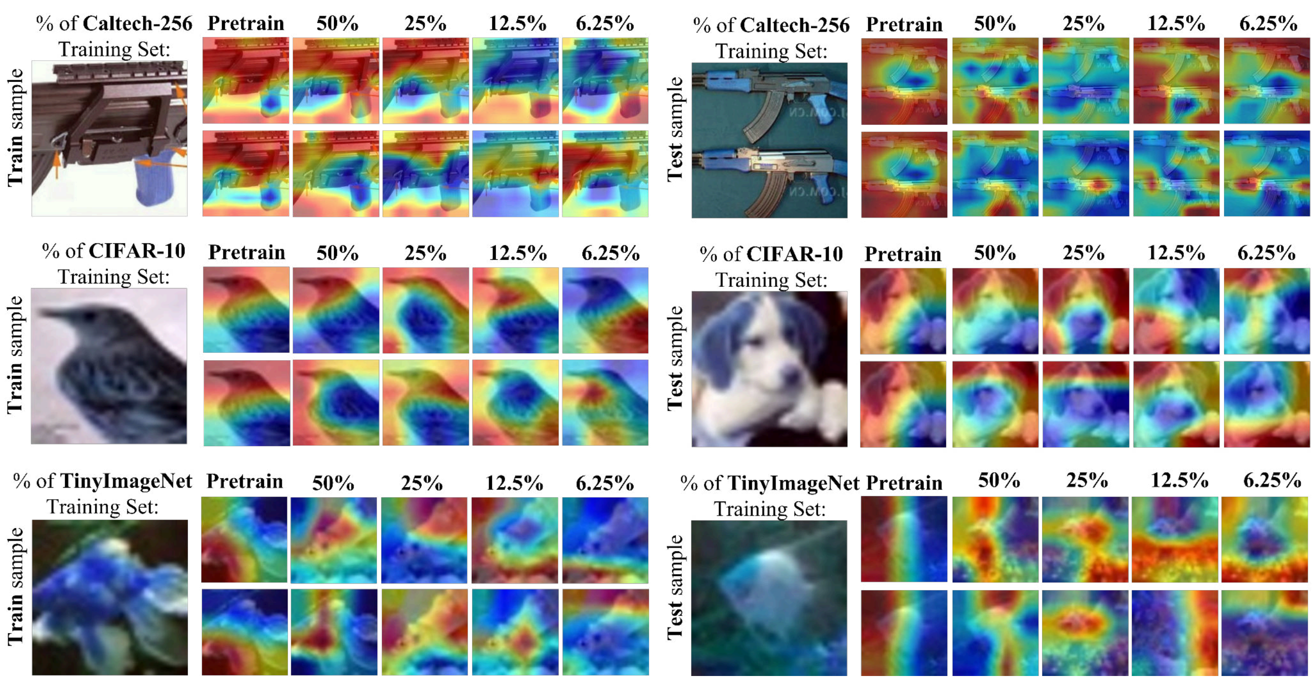


Figure 2: Grad-CAM comparisons with ResNet models. As we increase the number of training samples from 6.25% ($sf = 16$) of the training set, to using 50% of the training set, then to pre-training on ImageNet plus fine-tuning with 50% training data, we observe that model pairs generate increasingly consistent interpretations.

errors across measurements, datasets and network architectures increase. Furthermore, with increasing split factor, the correlation rank coefficient decreases; the saliency pixel intersection % decreases. These changes all point to the fact that training with fewer samples widens the interpretation generalization error, and worsens the disagreement between gradient-based interpretations. To further verify that generalization of network interpretations are more severely impacted than network predictions (by decreasing the training set size), in Figure 4, we plot the normalized Spearman correlation of interpretations against that of softmax predictions. We observe that with small and disjoint train sets, the quality and consistency of interpretation maps drops more rapidly than that of logit predictions.

2.2. Gradient-based Interpretations

We report the complete results on the SmoothGrad [7] regularization effect in Figures 5, 6, 7, 8, 9, 10, 11, 12. Across different network architectures and datasets, we verify that first, the average gradient norm difference increases as the number training samples decreases; subsequently, that SmoothGrad serves as a regularizer to decrease gradient difference and increase alignment; lastly, that increasing the number of Gaussian vectors m and increasing the noise standard deviation σ hyperparameters of SmoothGrad amplifies this regularization effect. This corroborates our message that more training data is needed for well-aligned gradients, and is consistent with visual results by Smilkov *et al*. [7], which demonstrated that SmoothGrad with increasing m, σ enhances the meaningfulness of gradient-based interpretations.

2.3. Interpretations in the Parameter Space

We present additional parameter space interpretations with CKA [3] in 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, where we evaluate the similarities of layer-wise representations of the same image across different models, as a function of train-set split size. We report detailed results using both linear and kernel CKA; darker colours indicate smaller self-similarities while lighter colors indicate greater self-similarities close to 1.0 (perfect agreement). As the split factor sf decreases, the cross-network representation self-similarities also decrease, meaning that interpretations are more inconsistent in the parameter space with fewer training data.

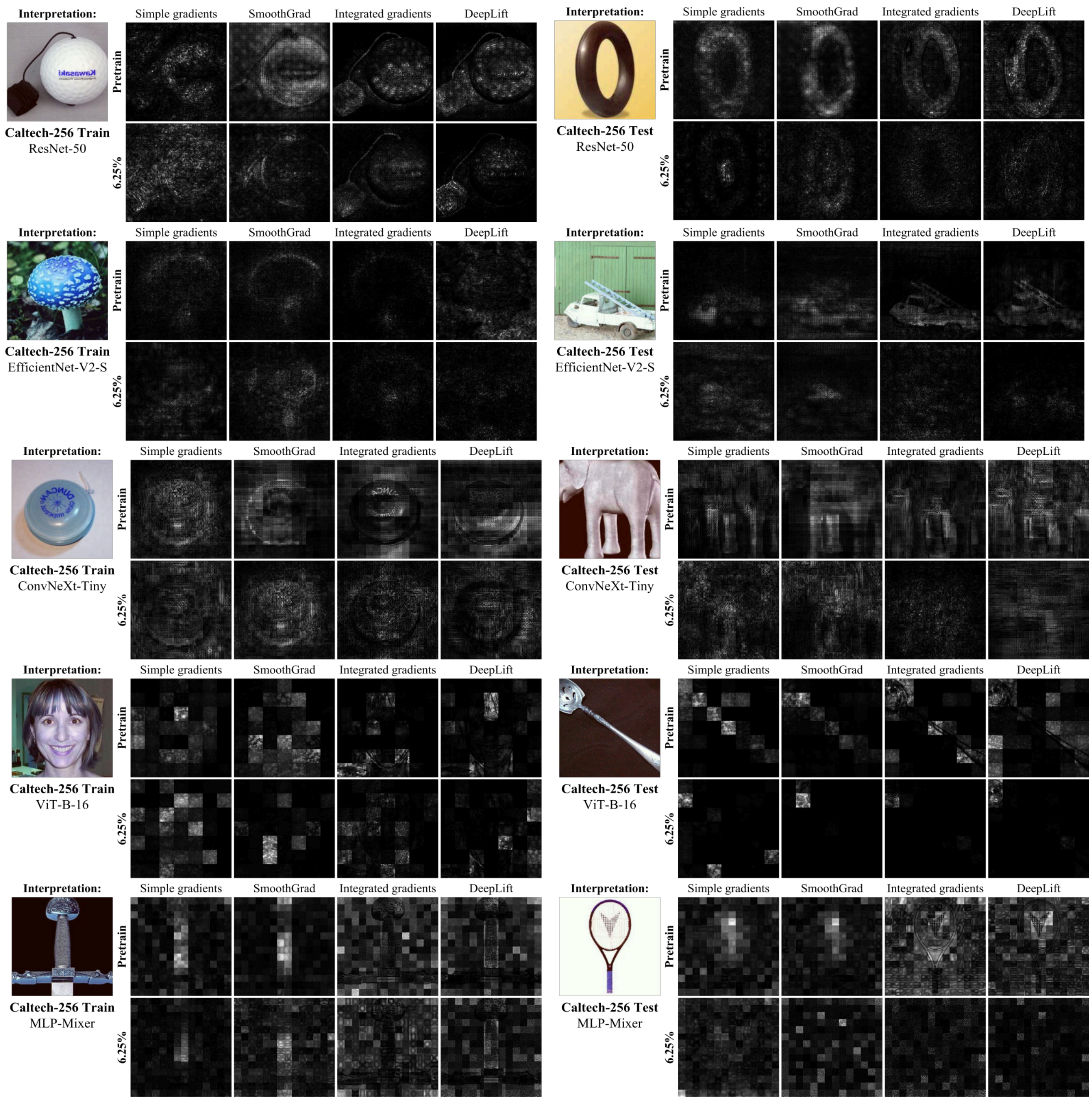
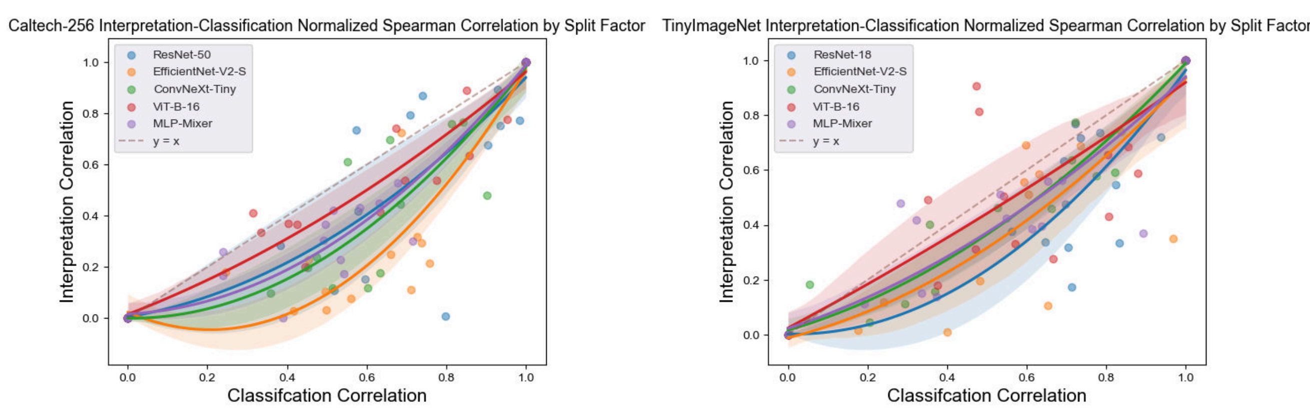


Figure 3: Different gradient-based interpretation methods tested on Caltech-256. We compare the fidelity, localization meaningfulness and train-test generalization abilities of interpretations, for Pretrain and the 6.25% experimental settings. We observe that the generalization and performance gaps widen for interpretations generated by models trained on smaller, disjoint training sets.

Rank Correlation & Saliency Pixel Intersection by Split Factor					
Dataset \ **Model**		EfficientNet-V2-S		ConvNeXt-Tiny	
		Rank C ↑	Px % ↑	Rank C ↑	Px % ↑
CIFAR-10	$sf = 2$	**.51** ± .04	**37.1** ± 2.5	**.43** ± .02	**37.9** ± 2.6
	$sf = 4$	.49 ± .03	33.1 ± 3.2	.41 ± .02	35.7 ± 2.0
	$sf = 8$	.38 ± .04	27.7 ± 1.9	.38 ± .02	34.6 ± 1.6
	$sf = 16$	.35 ± .03	26.4 ± 2.9	.36 ± .02	33.6 ± 1.4
Caltech-256	$sf = 2$	**.19** ± .01	**12.7** ± 2.0	**.25** ± .06	**12.3** ± 0.9
	$sf = 4$	.17 ± .04	9.5 ± 1.1	.22 ± .04	10.6 ± 1.6
	$sf = 8$	.15 ± .03	9.1 ± 1.0	.20 ± .08	9.6 ± 1.3
	$sf = 16$	.14 ± .03	8.4 ± 1.9	.14 ± .03	8.5 ± 1.2
TinyImageNet	$sf = 2$	**.25** ± .06	26.3 ± 2.4	**.11** ± .02	**23.6** ± 2.6
	$sf = 4$	.22 ± .04	**26.8** ± 3.3	.10 ± .03	23.3 ± 1.2
	$sf = 8$	.20 ± .08	23.8 ± 2.0	.08 ± .02	20.6 ± 1.0
	$sf = 16$	.14 ± .03	20.8 ± 3.5	.06 ± .01	20.1 ± 1.2

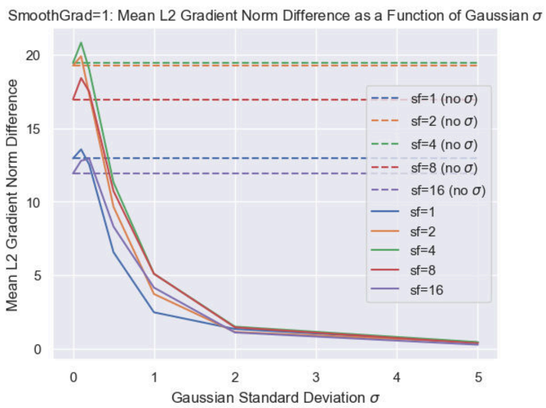
Table 1: Rank correlation coefficient and saliency pixel intersection on the test set, for the interpretations of neural nets trained with a training set split factor of $\text{sf} = 2, 4, 8, 16$.



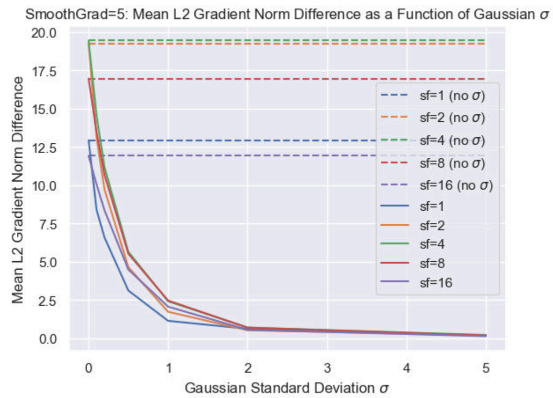
(a) Caltech-256 interpretation-classification correlations.

(b) TinyImageNet interpretation-classification correlations.

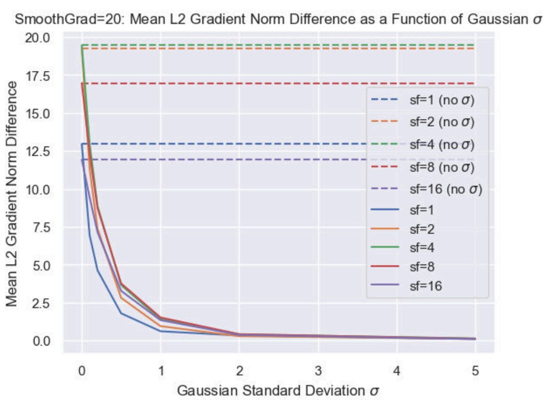
Figure 4: Normalized Spearman correlation of network interpretations against softmax predictions for Caltech-256 and TinyImageNet. As sf increases from 2 to 16 and models are trained with smaller disjoint train sets, the rank correlation of test set interpretations drop more acutely than that of network predictions.



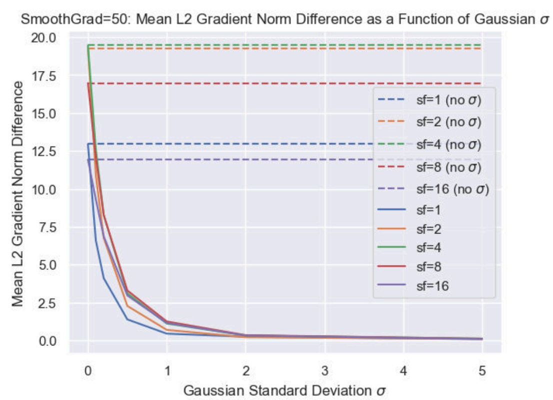
(a) $m = 1$ *(CIFAR-10, ResNet-18)*



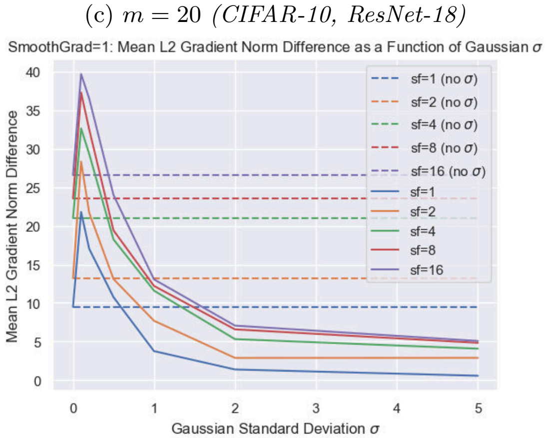
(b) $m = 5$ *(CIFAR-10, ResNet-18)*



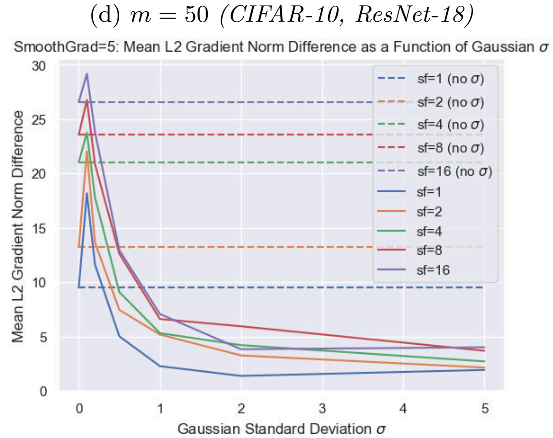
(c) $m = 20$ *(CIFAR-10, ResNet-18)*



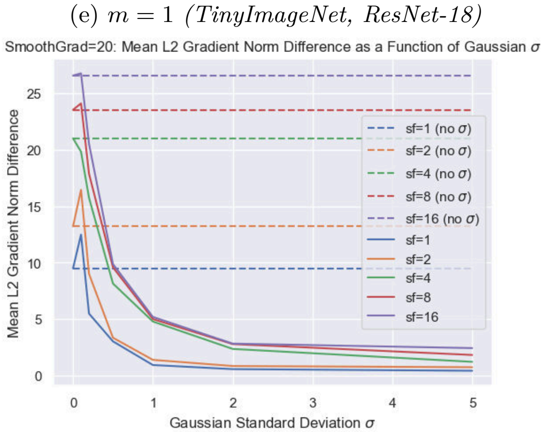
(d) $m = 50$ *(CIFAR-10, ResNet-18)*



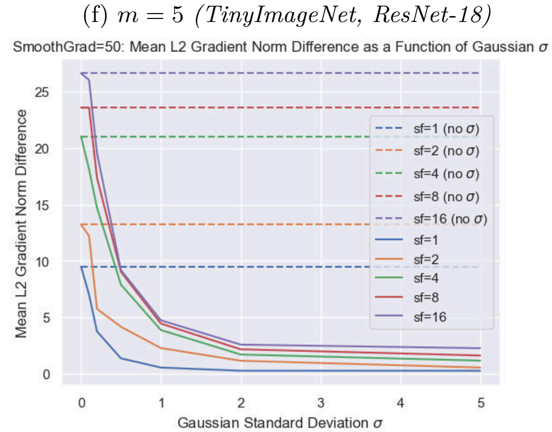
(e) $m = 1$ *(TinyImageNet, ResNet-18)*



(f) $m = 5$ *(TinyImageNet, ResNet-18)*

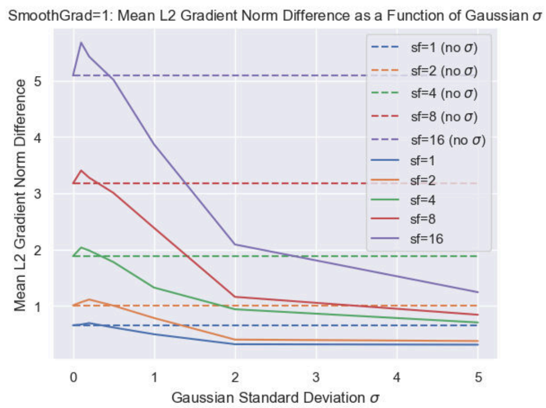


(g) $m = 20$ *(TinyImageNet, ResNet-18)*

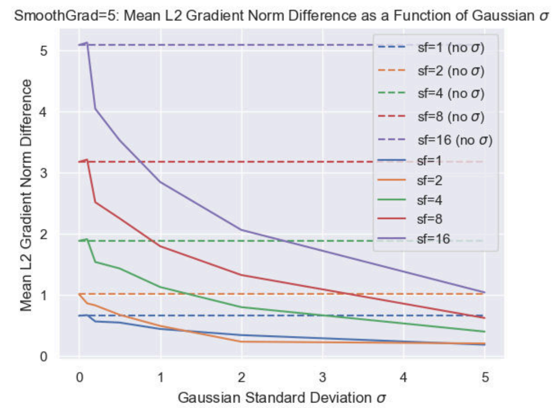


(h) $m = 50$ *(TinyImageNet, ResNet-18)*

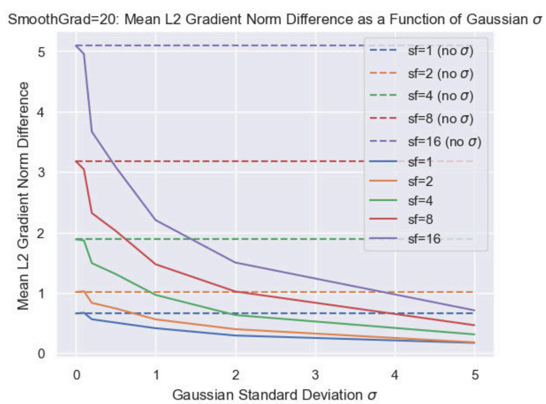
Figure 5



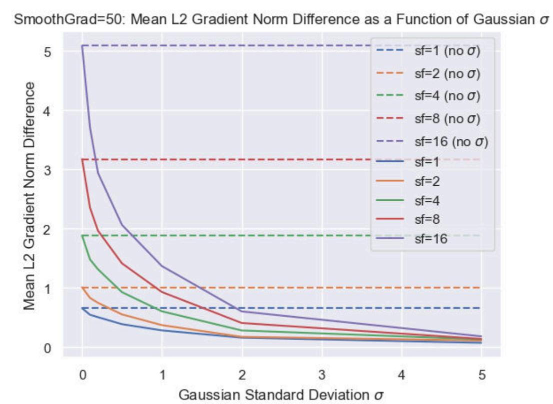
(a) $m = 1$ *(Caltech-256, ResNet-50)*



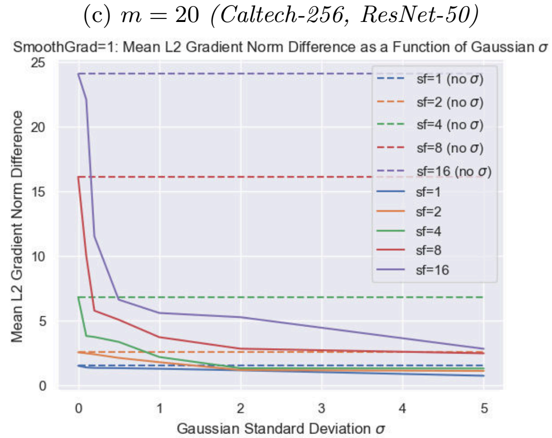
(b) $m = 5$ *(Caltech-256, ResNet-50)*



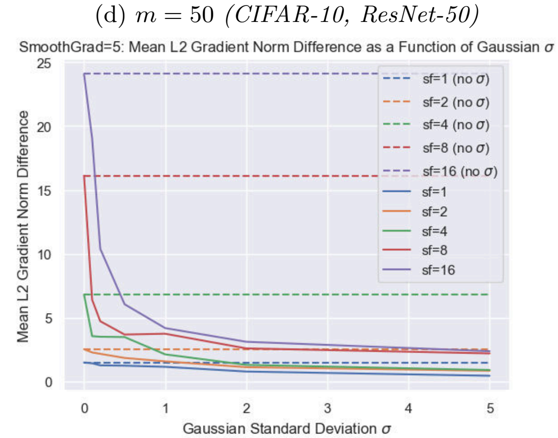
(c) $m = 20$ *(Caltech-256, ResNet-50)*



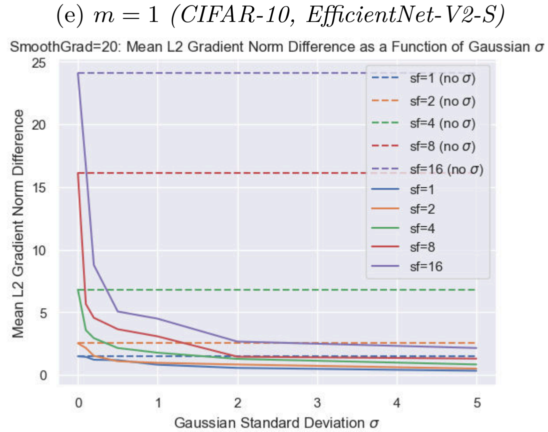
(d) $m = 50$ *(CIFAR-10, ResNet-50)*



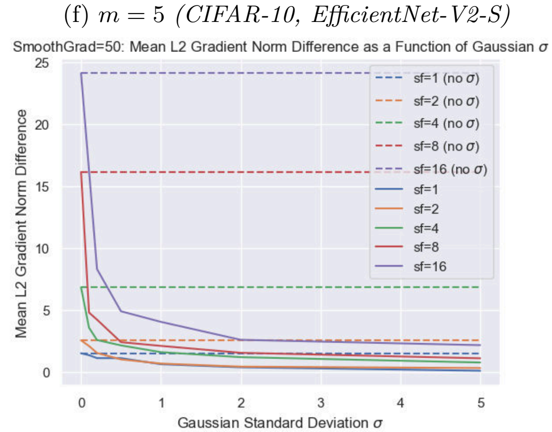
(e) $m = 1$ *(CIFAR-10, EfficientNet-V2-S)*



(f) $m = 5$ *(CIFAR-10, EfficientNet-V2-S)*

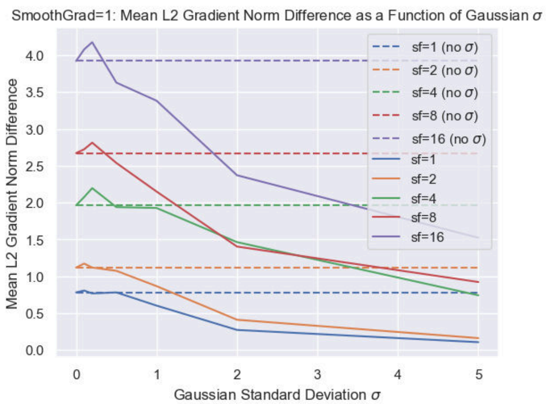


(g) $m = 20$ *(CIFAR-10, EfficientNet-V2-S)*

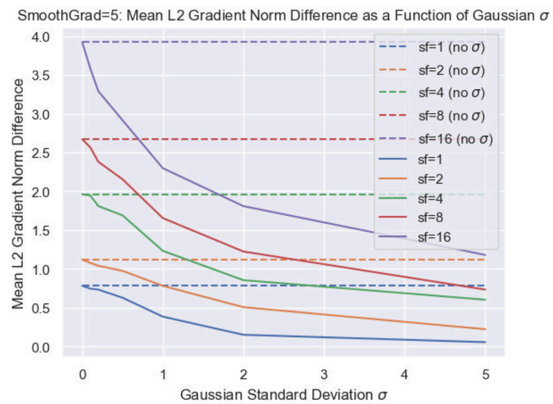


(h) $m = 50$ *(CIFAR-10, EfficientNet-V2-S)*

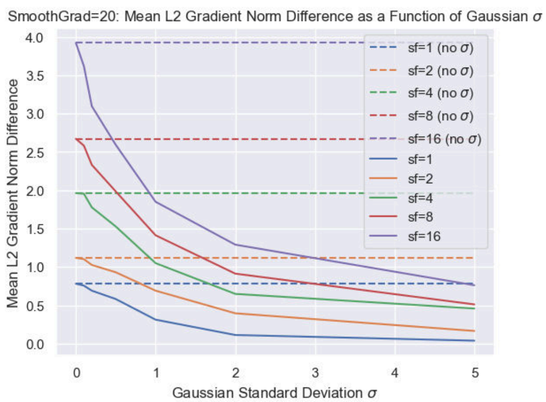
Figure 6



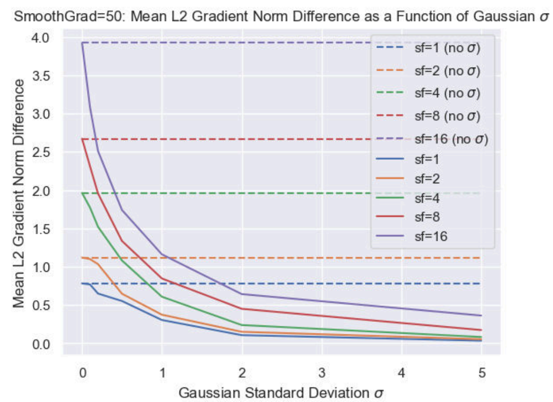
(a) $m = 1$ *(Caltech-256, EfficientNet-V2-S)*



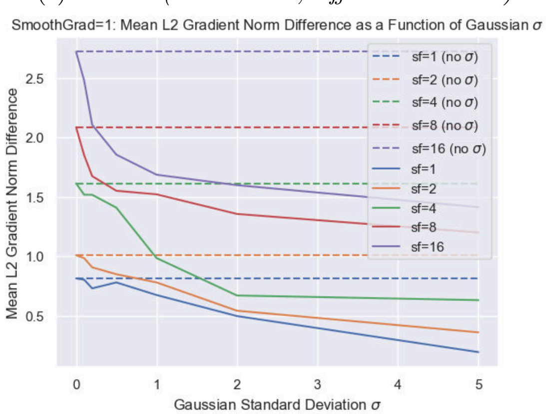
(b) $m = 5$ *(Caltech-256, EfficientNet-V2-S)*



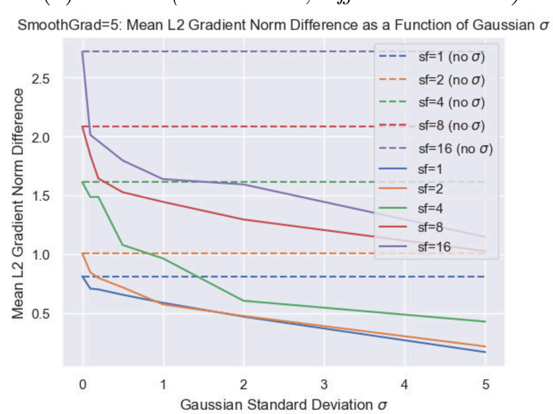
(c) $m = 20$ *(Caltech-256, EfficientNet-V2-S)*



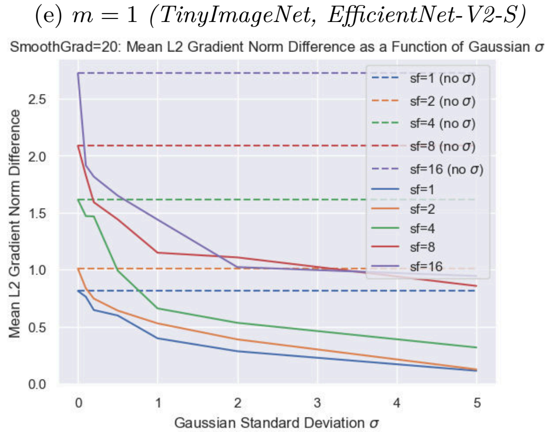
(d) $m = 50$ *(Caltech-256, EfficientNet-V2-S)*



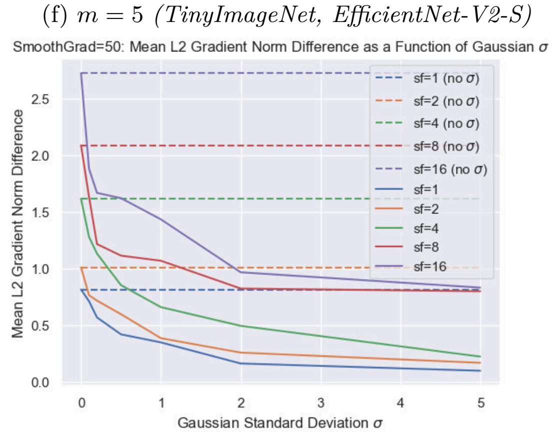
(e) $m = 1$ *(TinyImageNet, EfficientNet-V2-S)*



(f) $m = 5$ *(TinyImageNet, EfficientNet-V2-S)*

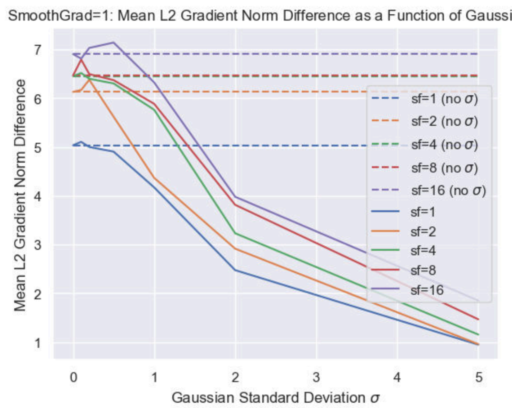


(g) $m = 20$ *(TinyImageNet, EfficientNet-V2-S)*

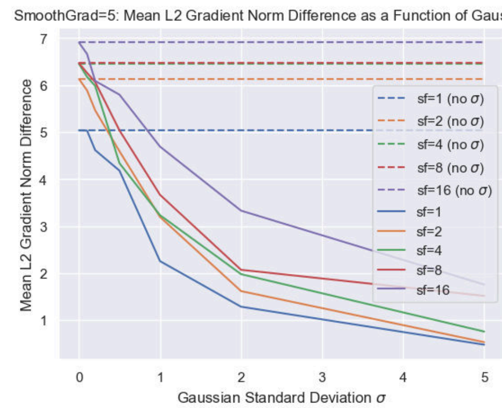


(h) $m = 50$ *(TinyImageNet, EfficientNet-V2-S)*

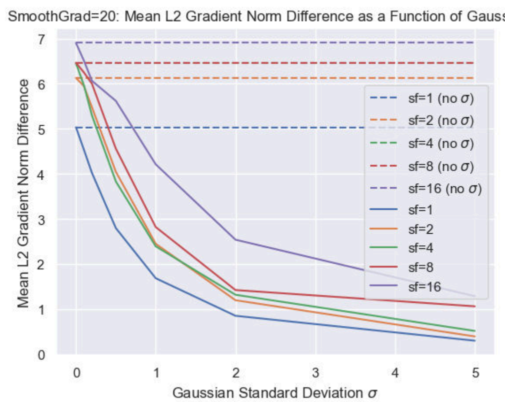
Figure 7



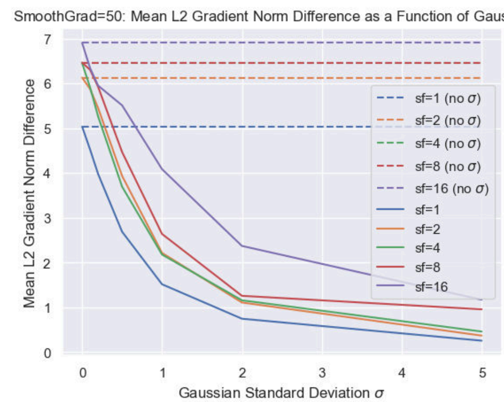
(a) $m = 1$ *(CIFAR-10, ConvNeXt-Tiny)*



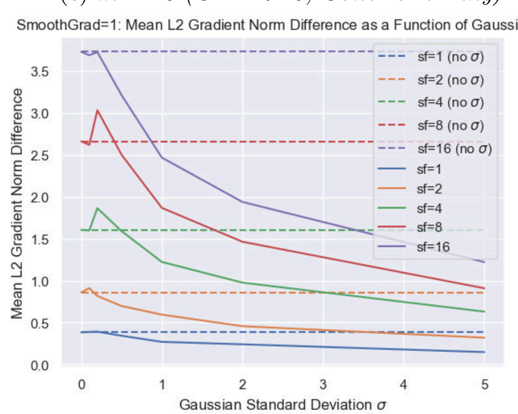
(b) $m = 5$ *(CIFAR-10, ConvNeXt-Tiny)*



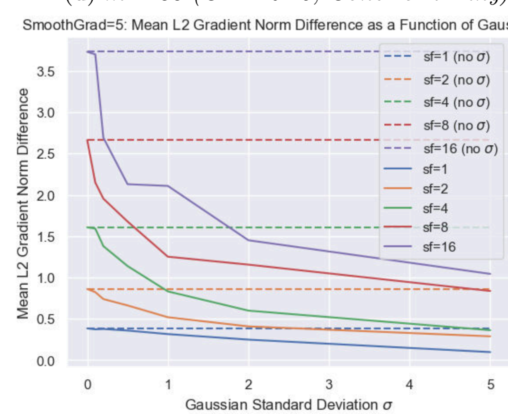
(c) $m = 20$ *(CIFAR-10, ConvNeXt-Tiny)*



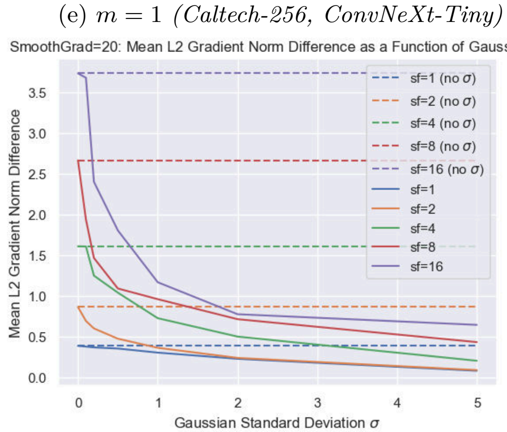
(d) $m = 50$ *(CIFAR-10, ConvNeXt-Tiny)*



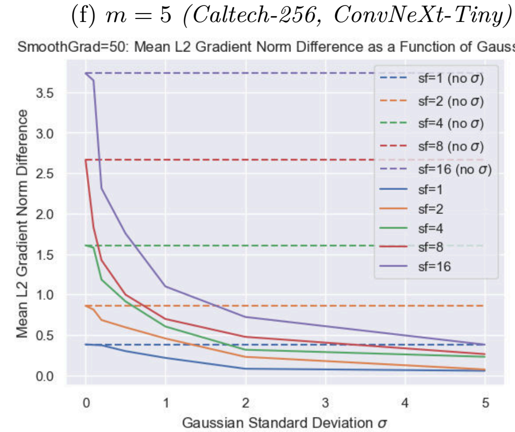
(e) $m = 1$ *(Caltech-256, ConvNeXt-Tiny)*



(f) $m = 5$ *(Caltech-256, ConvNeXt-Tiny)*

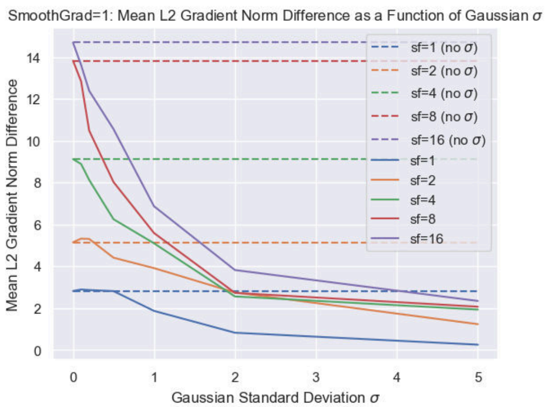


(g) $m = 20$ *(Caltech-256, ConvNeXt-Tiny)*

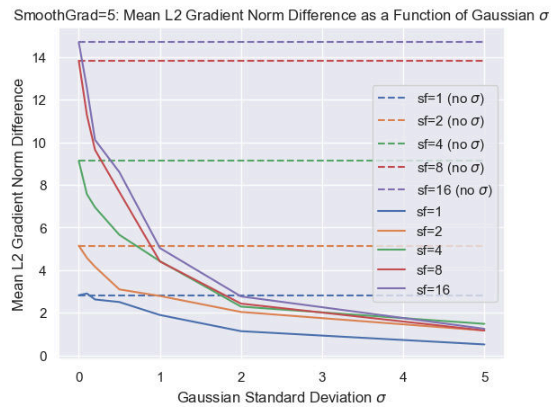


(h) $m = 50$ *(Caltech-256, ConvNeXt-Tiny)*

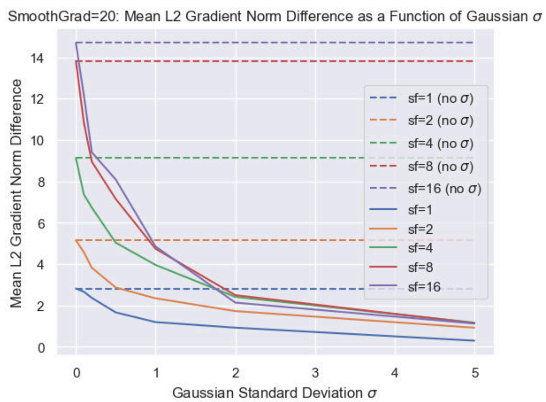
Figure 8



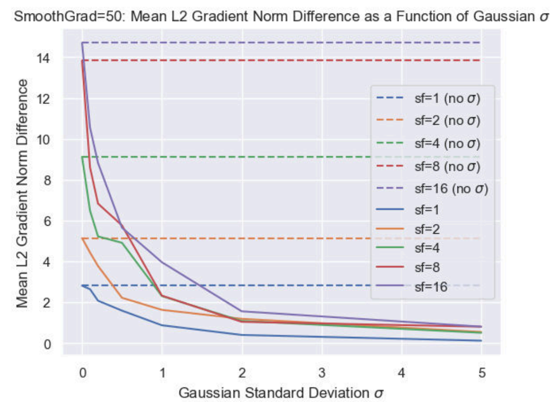
(a) $m = 1$ *(TinyImageNet, ConvNeXt-Tiny)*



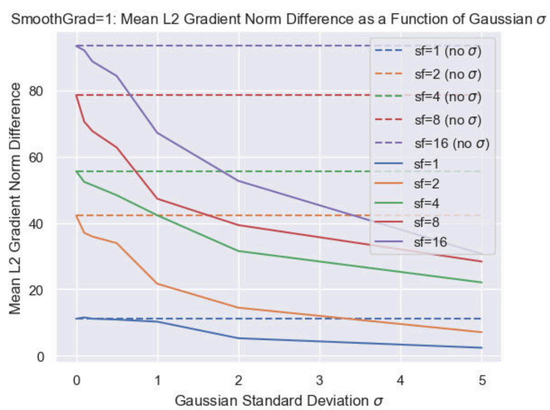
(b) $m = 5$ *(TinyImageNet, ConvNeXt-Tiny)*



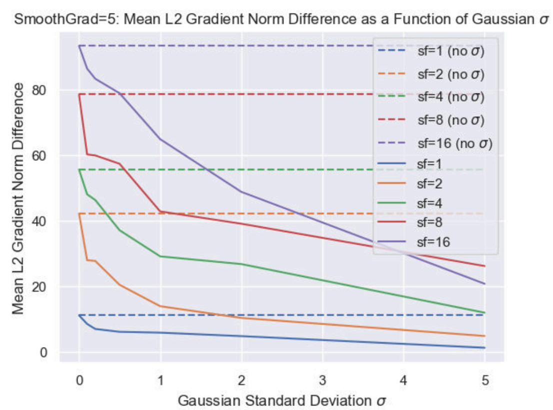
(c) $m = 20$ *(TinyImageNet, ConvNeXt-Tiny)*



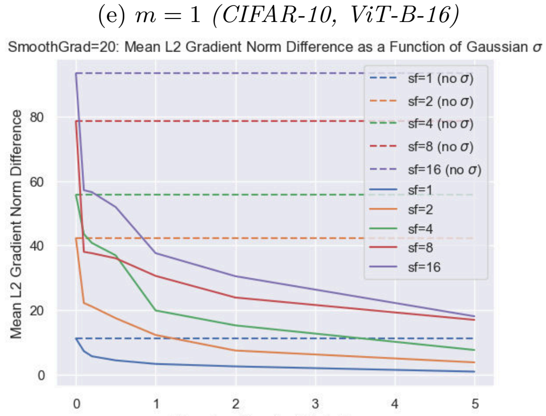
(d) $m = 50$ *(TinyImageNet, ConvNeXt-Tiny)*



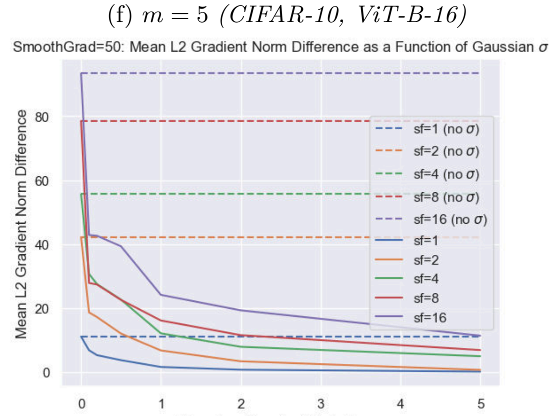
(e) $m = 1$ *(CIFAR-10, ViT-B-16)*



(f) $m = 5$ *(CIFAR-10, ViT-B-16)*

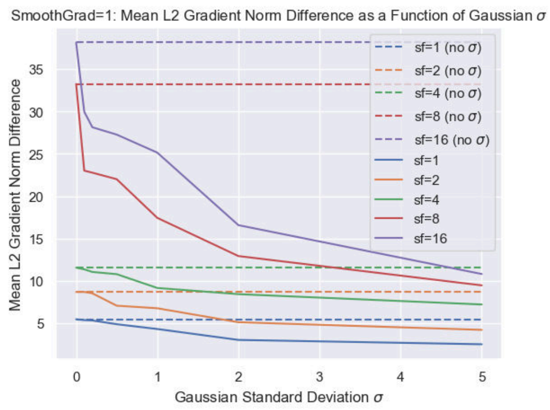


(g) $m = 20$ *(CIFAR-10, ViT-B-16)*

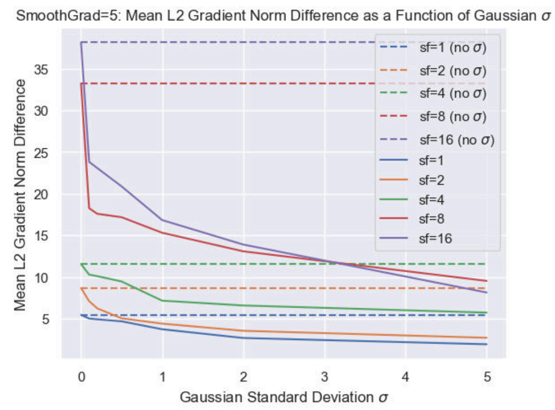


(h) $m = 50$ *(CIFAR-10, ViT-B-16)*

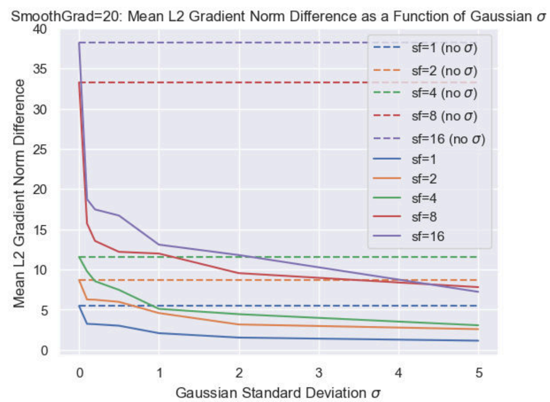
Figure 9



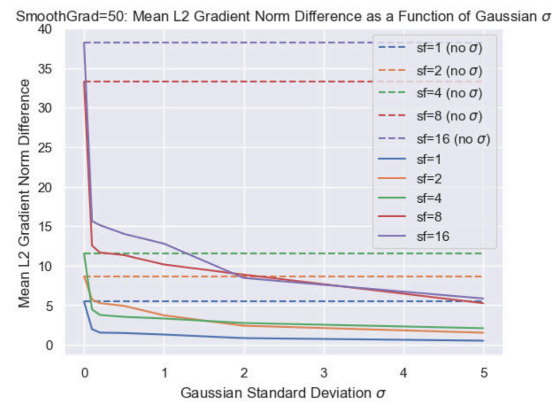
(a) $m = 1$ *(Caltech-256, ViT-B-16)*



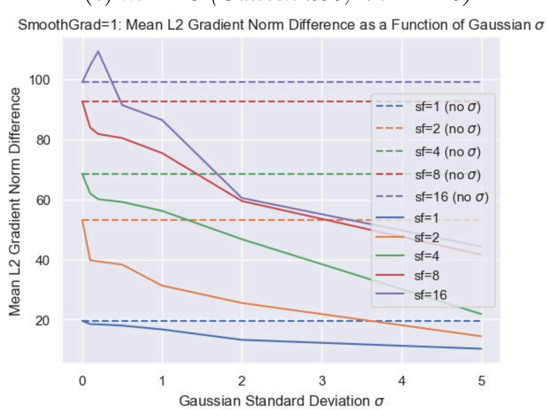
(b) $m = 5$ *(Caltech-256, ViT-B-16)*



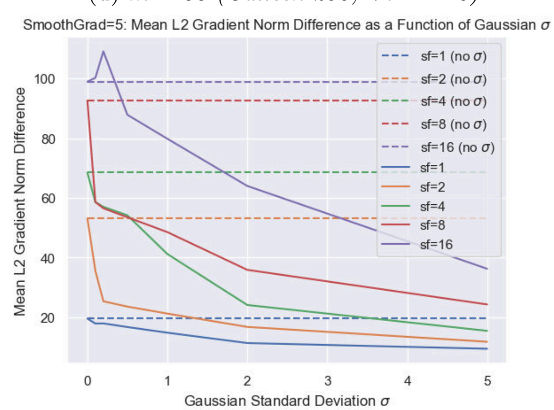
(c) $m = 20$ *(Caltech-256, ViT-B-16)*



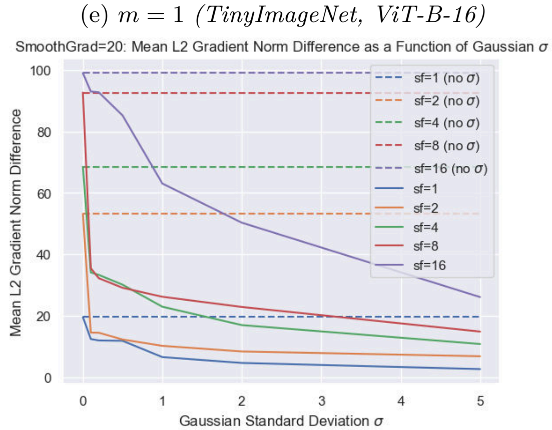
(d) $m = 50$ *(Caltech-256, ViT-B-16)*



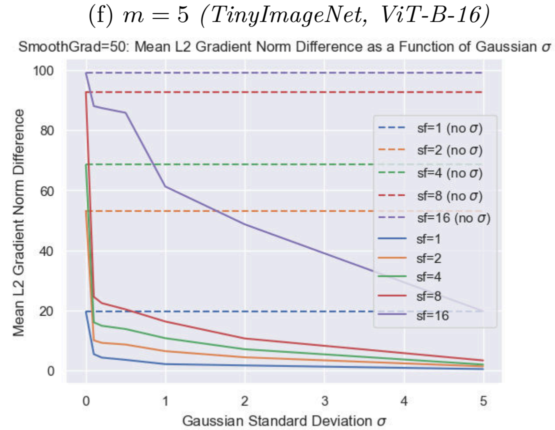
(e) $m = 1$ *(TinyImageNet, ViT-B-16)*



(f) $m = 5$ *(TinyImageNet, ViT-B-16)*

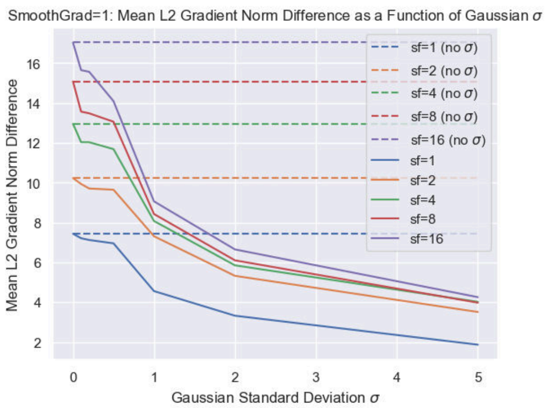


(g) $m = 20$ *(TinyImageNet, ViT-B-16)*

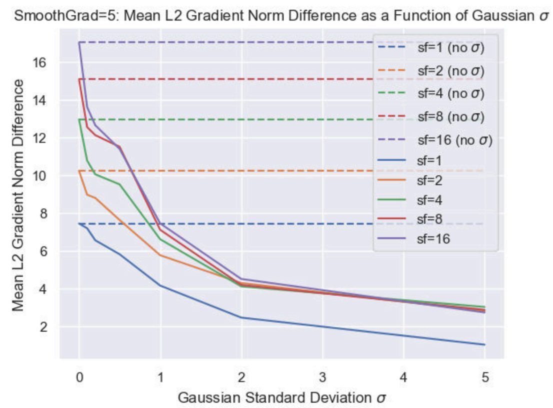


(h) $m = 50$ *(TinyImageNet, ViT-B-16)*

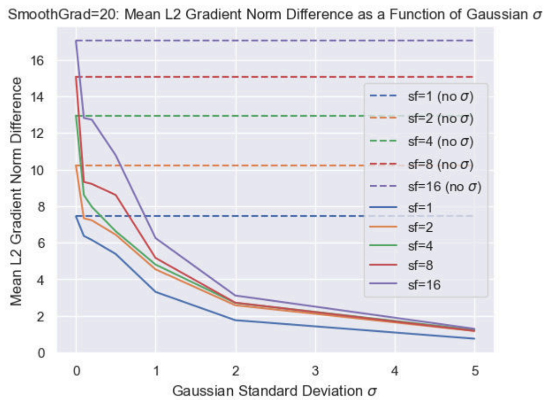
Figure 10



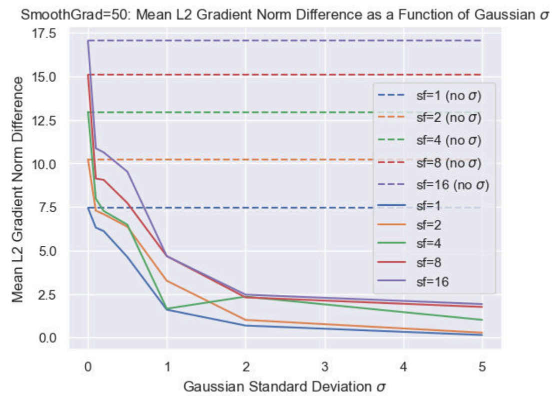
(a) $m = 1$ *(CIFAR-10, MLP-Mixer)*



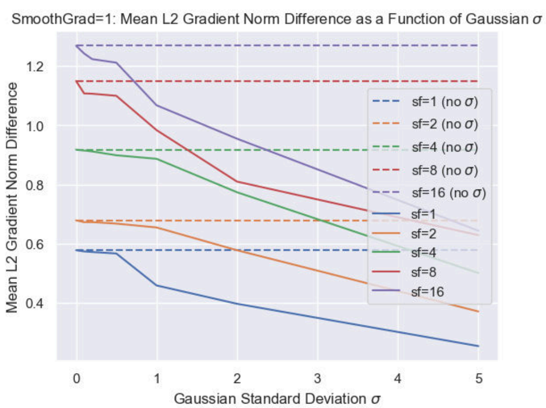
(b) $m = 5$ *(CIFAR-10, MLP-Mixer)*



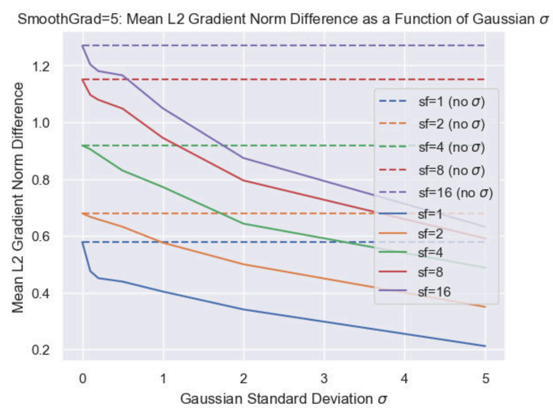
(c) $m = 20$ *(CIFAR-10, MLP-Mixer)*



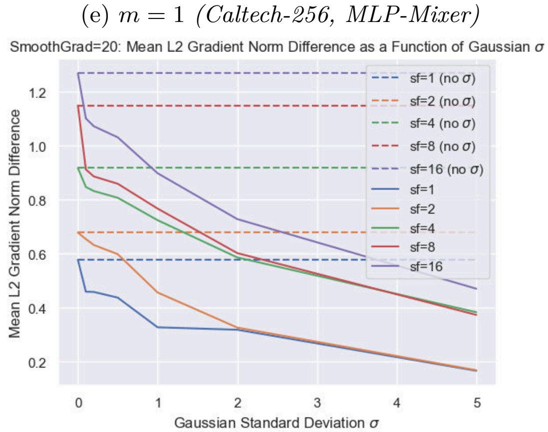
(d) $m = 50$ *(CIFAR-10, MLP-Mixer)*



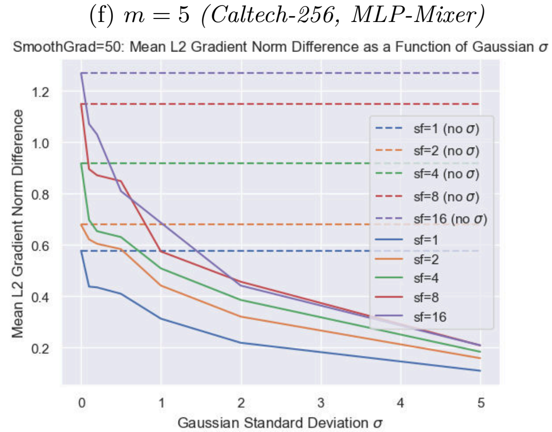
(e) $m = 1$ *(Caltech-256, MLP-Mixer)*



(f) $m = 5$ *(Caltech-256, MLP-Mixer)*

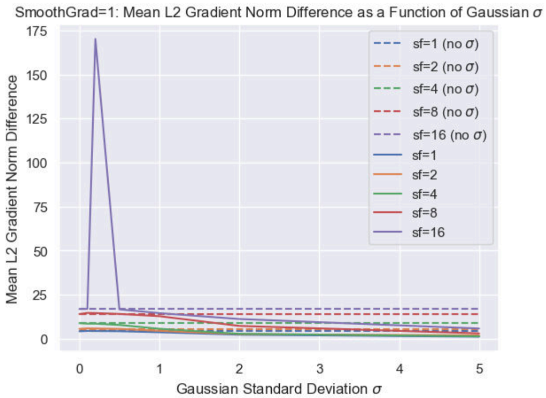


(g) $m = 20$ *(Caltech-256, MLP-Mixer)*

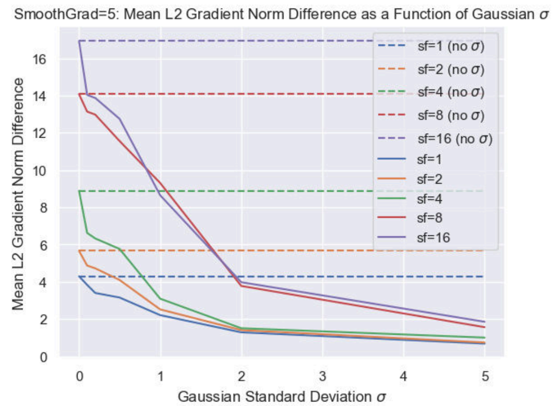


(h) $m = 50$ *(Caltech-256, MLP-Mixer)*

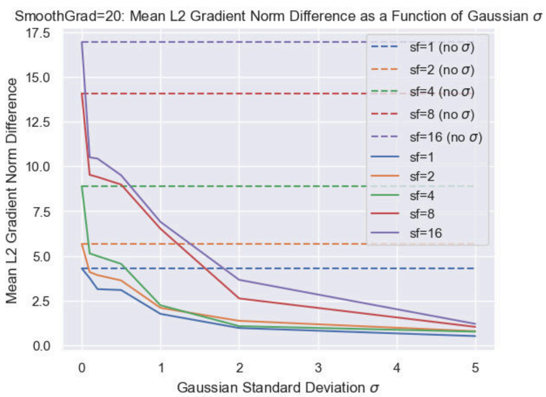
Figure 11



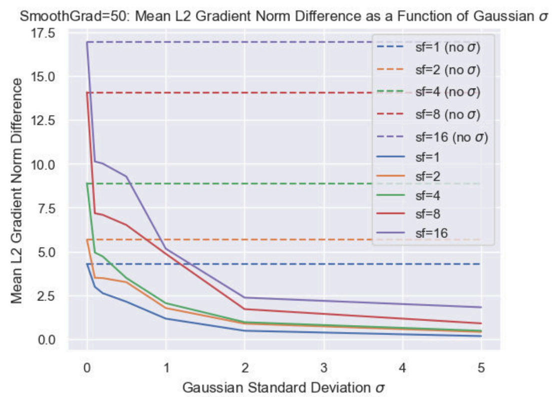
(a) $m = 1$ *(TinyImageNet, MLP-Mixer)*



(b) $m = 5$ *(TinyImageNet, MLP-Mixer)*

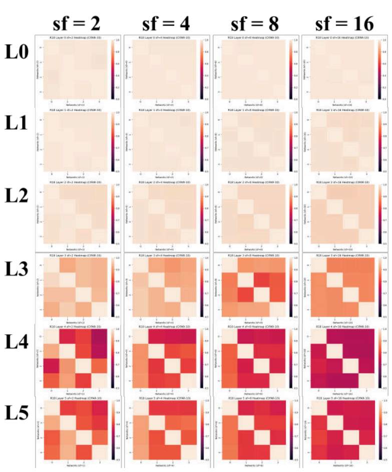


(c) $m = 20$ *(TinyImageNet, MLP-Mixer)*

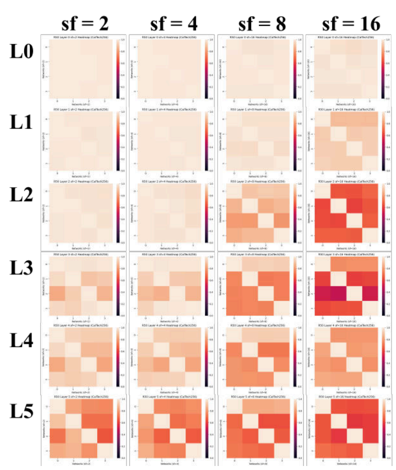


(d) $m = 50$ *(TinyImageNet, MLP-Mixer)*

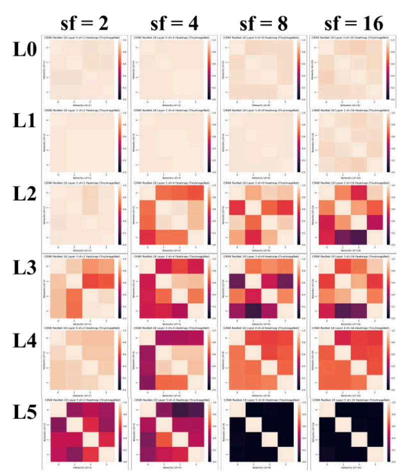
Figure 12



(a) CIFAR-10, ResNet-18

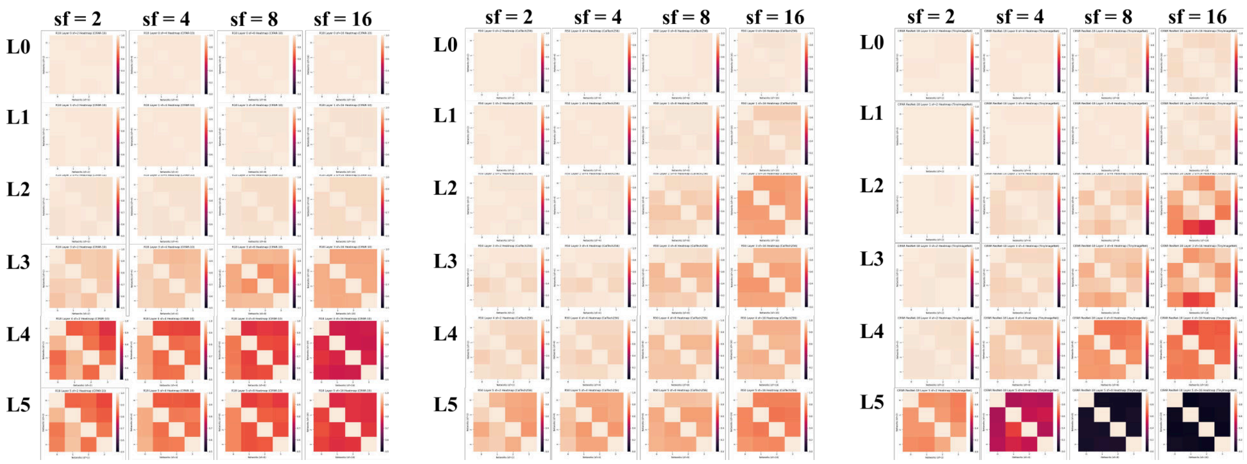


(b) Caltech-256, ResNet-50



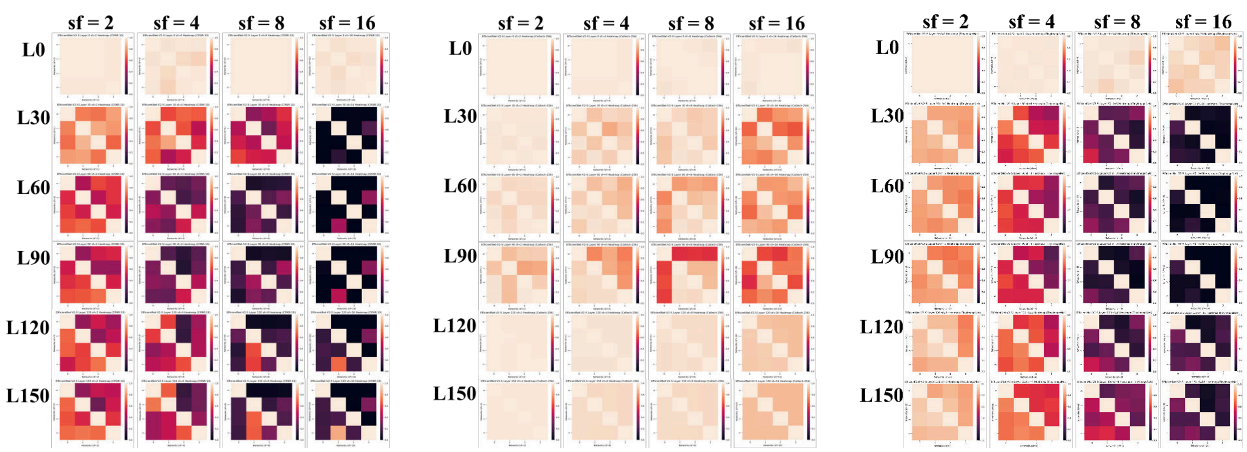
(c) TinyImageNet, ResNet-18

Figure 13: **Linear** CKA interpretation comparisons for 6 layers of ResNet-18 and 50.



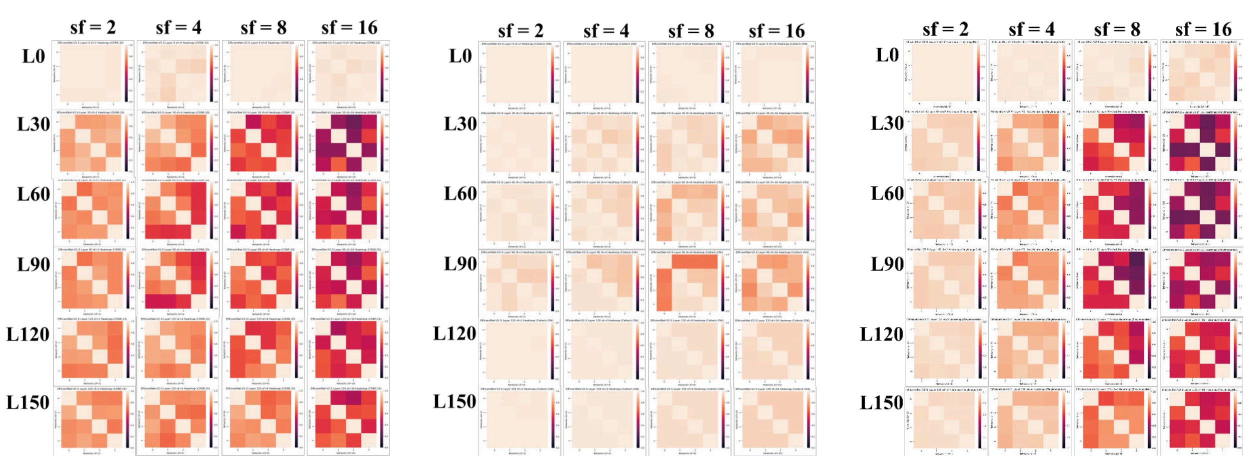
(a) CIFAR-10, ResNet-18 (b) Caltech-256, ResNet-50 (c) TinyImageNet, ResNet-18

Figure 14: **Kernel** CKA interpretation comparisons for 6 layers of ResNet-18 and 50.



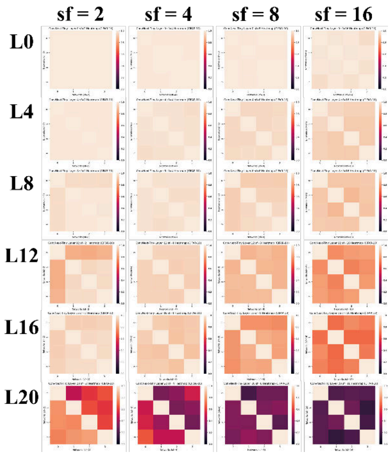
(a) CIFAR-10, EfficientNet-V2-S (b) Caltech-256, EfficientNet-V2-S (c) TinyImageNet, EfficientNet-V2-S

Figure 15: **Linear** CKA interpretation comparisons for 6 layers of EfficientNet-V2-S.

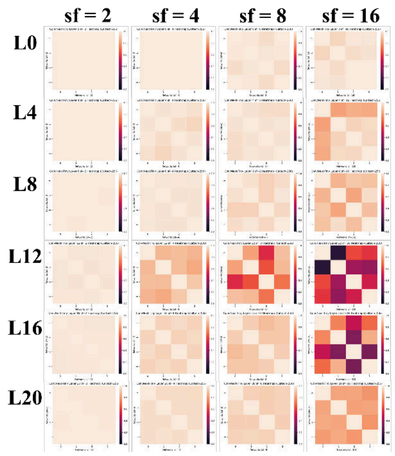


(a) CIFAR-10, EfficientNet-V2-S (b) Caltech-256, EfficientNet-V2-S (c) TinyImageNet, EfficientNet-V2-S

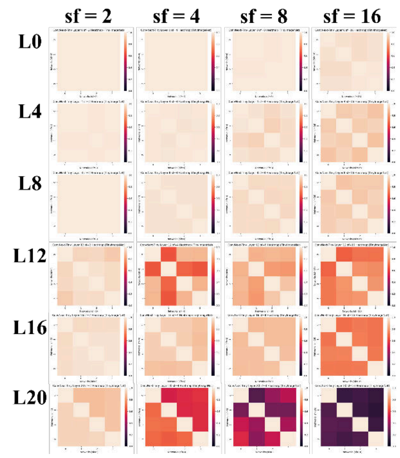
Figure 16: **Kernel** CKA interpretation comparisons for 6 layers of EfficientNet-V2-S.



(a) CIFAR-10, ConvNeXt-Tiny

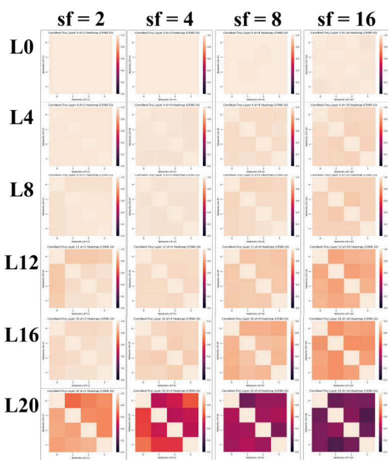


(b) Caltech-256, ConvNeXt-Tiny

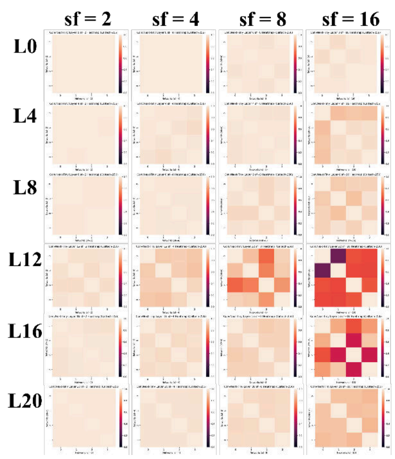


(c) TinyImageNet, ConvNeXt-Tiny

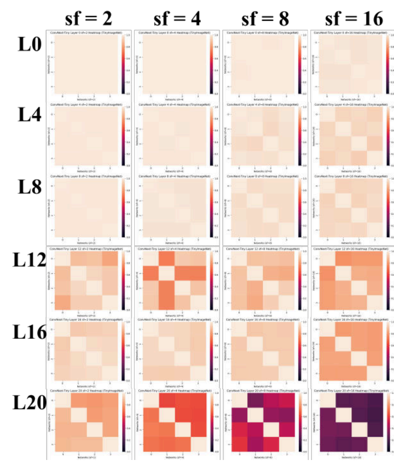
Figure 17: **Linear** CKA interpretation comparisons for 6 layers of ConvNeXt-Tiny.



(a) CIFAR-10, ConvNeXt-Tiny

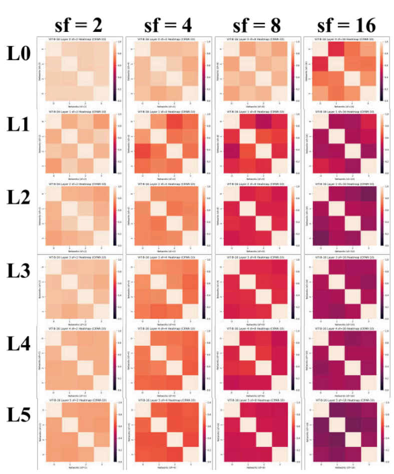


(b) Caltech-256, ConvNeXt-Tiny

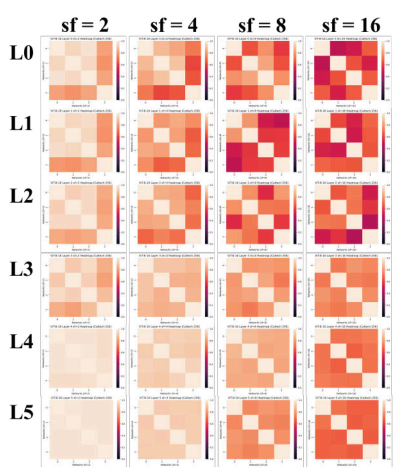


(c) TinyImageNet, ConvNeXt-Tiny

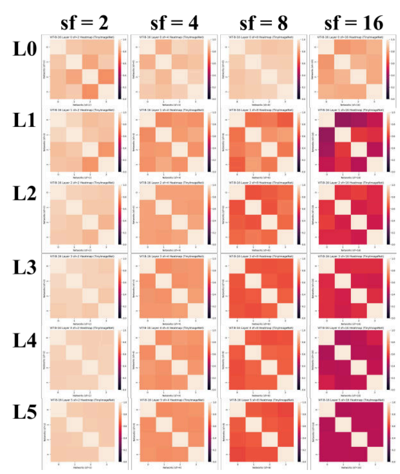
Figure 18: **Kernel** CKA interpretation comparisons for 6 layers of ConvNeXt-Tiny.



(a) CIFAR-10, ViT-B-16

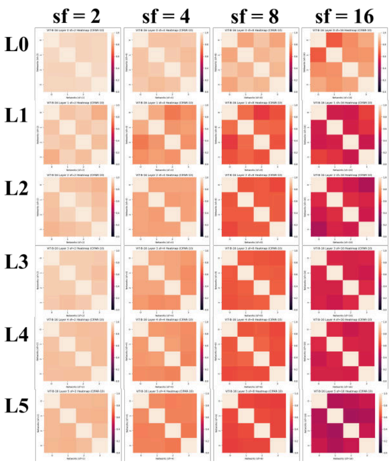


(b) Caltech-256, ViT-B-16

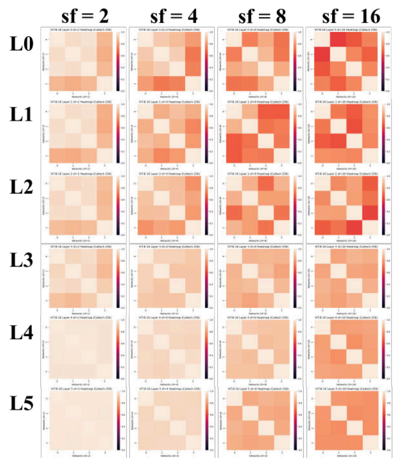


(c) TinyImageNet, ViT-B-16

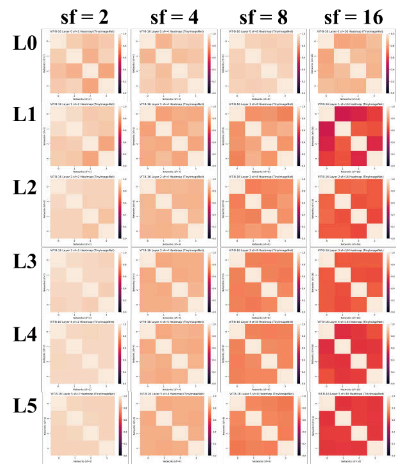
Figure 19: **Linear** CKA interpretation comparisons for 6 layers of ViT-B-16.



(a) CIFAR-10, ViT-B-16

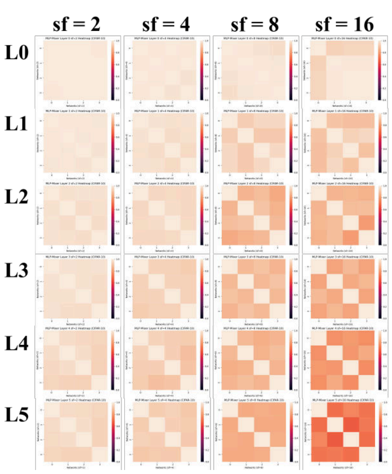


(b) Caltech-256, ViT-B-16

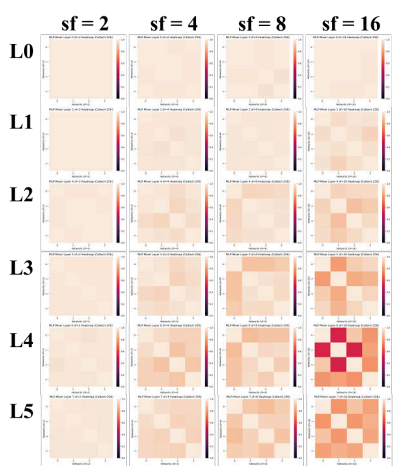


(c) TinyImageNet, ViT-B-16

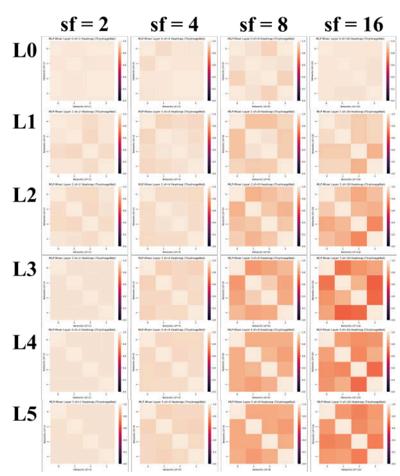
Figure 20: **Kernel** CKA interpretation comparisons for 6 layers of ViT-B-16.



(a) CIFAR-10, MLP-Mixer

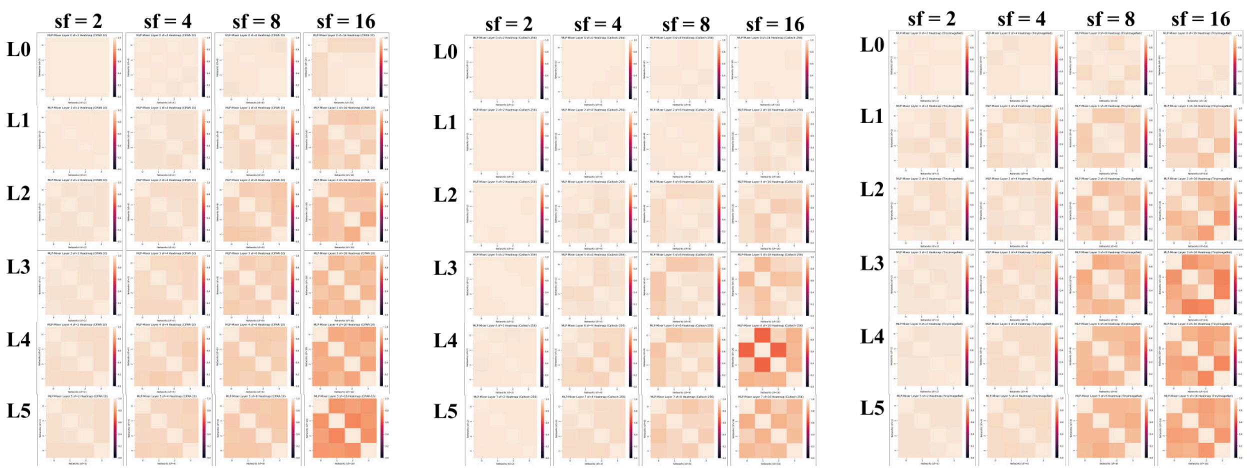


(b) Caltech-256, MLP-Mixer



(c) TinyImageNet, MLP-Mixer

Figure 21: **Linear** CKA interpretation comparisons for 6 layers of MLP-Mixer.



(a) CIFAR-10, MLP-Mixer (b) Caltech-256, MLP-Mixer (c) TinyImageNet, MLP-Mixer

Figure 22: **Kernel** CKA interpretation comparisons for 6 layers of MLP-Mixer.

References

[1] Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. *Advances in neural information processing systems*, 30, 2017. 1, 2, 3, 4

[2] Griffin, Holub, and Perona. Caltech 256, Apr 2022. 4

[3] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pages 3519–3529. PMLR, 2019. 6

[4] 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 *CVPR*, pages 618–626, 2017. 4

[5] 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. 4

[6] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013. 4

[7] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017. 4, 6

[8] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR, 2017. 4

[9] Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018. 3