SMOOT: Saliency Guided Mask Optimized Online Training

1st Ali Karkehabadi
Department of ECE
University of California, Davis
Davis, California, USA
akarkehabadi@ucdavis.edu

2nd Houman Homayoun
Department of ECE
University of California, Davis
Davis, California, USA
hhomayoun@ucdavis.edu

3rd Avesta Sasan
Department of ECE
University of California, Davis
Davis, California, USA
asasan@ucdavis.edu

2024 IEEE 17th Dallas Circuits and Systems Conference (DCAS) | 979-8-3503-4953-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/DCAS61159.2024.10539909

***Abstract*—Deep Neural Networks are powerful tools for understanding complex patterns and making decisions. However, their black-box nature impedes a complete understanding of their inner workings. Saliency-Guided Training (SGT) methods try to highlight the prominent features in the model's training based on the output to alleviate this problem. These methods use back-propagation and modified gradients to guide the model toward the most relevant features while keeping the impact on the prediction accuracy negligible. SGT makes the model's final result more interpretable by masking input partially. In this way, considering the model's output, we can infer how each segment of the input affects the output. In the particular case of image as the input, masking is applied to the input pixels. However, the masking strategy and number of pixels which we mask, are considered as a hyperparameter. Appropriate setting of masking strategy can directly affect the model's training. In this paper, we focus on this issue and present our contribution. We propose a novel method to determine the optimal number of masked images based on input, accuracy, and model loss during the training. The strategy prevents information loss which leads to better accuracy values. Also, by integrating the model's performance in the strategy formula, we show that our model represents the salient features more meaningful. Our experimental results demonstrate a substantial improvement in both model accuracy and the prominence of saliency, thereby affirming the effectiveness of our proposed solution.**

***Index Terms*—Deep Learning, Saliency Guided Training, Interpretability, Masking Strategy, Model Improvement**

I. Introduction

The transformative influence stems from its ability to learn from data and discover complex patterns in complex datasets. Deep Neural Networks (DNNs) have revolutionized prediction accuracy, yet their opaque nature raises concerns about reliability. Understanding DNN behavior is crucial, especially in sensitive fields like medicine, neuroscience, finance, and autonomous driving [1]. This understanding aids in model debugging and tuning. Research has focused on interpretability methods, including identifying influential input features for classification decisions [4]. Commonly referred to as saliency maps, these methods typically employ gradient calculations to assign an importance score to individual features, thus reflecting their impacts on the model's prediction [4]. Saliency maps can be unclear due to noise or distracting elements, which makes less accurate. To address this issue, [13] proposed explanation methods that leverage higher-order backward gradients to give insight into the saliency maps. An example is the SmoothGrad technique, which mitigates saliency noise by repetitively adding noise to the input and subsequently averaging the resulting saliency maps for each input [17]. Other techniques like DeepLIFT [4], and Layer-wise Relevance Propagation [15] modify the backpropagation through a different gradient function [12]. However, these methods' effectiveness is intrinsically tied to their reliability and stability [11]. If saliency maps change drastically for slight perturbations in the input or model, their trustworthiness can be severely compromised [3]. Thus, in developing novel interpretability techniques, it is imperative to establish robust and comprehensive sanity checks to ensure their validity and [11]. Furthermore, the quality of explanations generated by these methods can vary significantly depending on the data type (images, text, time series, etc.) and the model architecture (CNN, Recurrent Neural Networks, Transformer-based models, etc.). Hence, it's crucial to develop new interpretation techniques considering these factors [10]. Moreover, the quest for better interpretability extends beyond understanding individual predictions. It's about deciphering the learned representations and the model's decision-making logic [2]. Neural network distillation into interpretable models like soft decision trees has been studied as a means to improve interoperability [8]. This paper extends existing gradient-based methods to better understand model behavior and improve generalization by selecting robust features during training. We review relevant literature on interpretability and saliency-guided training to build upon prior works and enhance the effectiveness of gradient-based approaches.

A. Interpretability

Recent studies have introduced various methods to enhance neural network interpretability. Perkins et al. developed a feature selection grafting technique, optimizing the training process for large datasets [16]. Ghaeini et al. focused on saliency learning to align model explanations with actual ground truths [9]. Wang et al. emphasized class discrimination in training CNNs to improve accuracy and reduce visual confusion [19]. DeVries et al. demonstrated the effectiveness

979-8-3503-4953-5/24/$31.00 ©2024 IEEE

of cutout regularization in enhancing CNN robustness and performance [5].

B. Saliency Guided Training

In the saliency-guided training (SGT), Ismail et al. [18] introduce a new algorithm incorporating interpretability to enhance models' accuracy and saliency. Algorithm 1 describes the SGT process which uses saliency information in training a neural network model f_θ. In this algorithm $\mathcal{D}_{\mathcal{KL}}(p\|q)$ is the KL divergence between probability distributions p and q. the $\mathcal{D}_{\mathcal{KL}}$ quantifies the difference between the original output distribution $f_\theta(X)$ and the modified output distribution $f_\theta(\widetilde{X})$. The $M_k(I, X)$ is the masking function that removes the bottom k features from the input data X, based on the sorted index I representing the importance of features according to their gradients. The $\widetilde{X}$ is the input data with the least important k features masked out. It is obtained by applying the masking operation $M_k(I, X)$. The L_i is the combined loss function used for training. It includes two terms: the standard loss term $\mathcal{L}(f_\theta(X), y)$ that measures the model's performance on the original input X with corresponding labels y, and a regularization term involving the KL divergence to encourage similarity between the output distributions of X and $\widetilde{X}$.

Algorithm 1 Saliency Guided Training (Original)

Training samples X, number of features to be masked k, learning rate τ, hyperparameter λ
Initialize f_θ {Preload or randomize for new training}
for $i = 1$ ***to*** *epochs* **do**
  for *minibatch* **do**
    {Calculate the sorted index I for the gradient of output w.r.t the input.}
    $I = \text{sort}(\nabla_X f_{\theta_i}(X))$
    {Mask the bottom k features of the original input.}
    $\widetilde{X} = M_k(I, X)$
    {Compute the loss function with regularization term.}
    $L_i = \mathcal{L}(f_{\theta_i}(X), y) + \lambda \mathcal{D}_{\mathcal{KL}}(f_{\theta_i}(X) \| f_{\theta_i}(\widetilde{X}))$
    {Update network parameters using the gradient.}
    $f_{\theta_{i+1}} = f_{\theta_i} - \tau \nabla_{\theta_i} L_i$
  end
end

C. Motivation and Problem Statement

In Algorithm 1, parameter k determines the number of masked features, which is considered constant despite its potential impact on SGT optimization as noted by [18]. Our investigation into k's influence, realizing its optimal value varies with the input image, led to our proposed solution. We computed gradients for all input pixels via backpropagation, ranked them, and began masking from the highest gradients, anticipating a decline in model accuracy with additional masking. However, some images showed an initial accuracy increase upon masking high gradients, peaking before decreasing. This phenomenon is depicted in Figure 1, where the orange curve shows the expected accuracy decline with increased masking for most images, and the blue curve indicates the unusual cases of peaking accuracy, occurring before or after masking 50

The study investigated the impact of feature masking on model accuracy using a two-layer convolutional neural network (CNN) trained on the CIFAR-10 dataset. The findings showed that 16% of test images experienced increased accuracy after masking, indicating the significance of this phenomenon. Specific image characteristics influence the optimal masking parameter (k), affecting which pixels are crucial for the model's decision-making. Saliency maps supported these observations, highlighting attention given to irrelevant pixels. Based on this, the study proposes optimizing k using saliency metrics to enhance feature learning and align saliency maps with objects of interest, addressing the problem of formulating a saliency-guided training solution to improve model accuracy and saliency map fidelity.

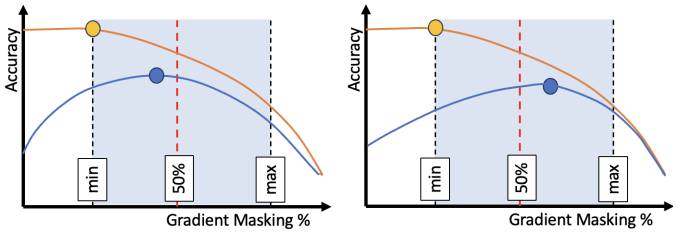

Fig. 1. Illustration of how sorted gradient masking could result in a monotonic decrease in accuracy in the majority of images (in orange), but an initial increase and then decrease in accuracy in some other images (in blue). The figure on the left captures the case where the peak accuracy in the exception images is reached before the 50% masking point, and the figure on the right captures the case where peak accuracy in the exception images is reached after 50% masking point.

II. Methodology

To improve model generalization and saliency map accuracy by emphasizing key features and reducing irrelevant noise, we developed the Saliency Guided Mask Optimized Online Training (SMOOT) method. This technique dynamically adjusts the hyperparameter k, which dictates the count of masked pixels, enhancing input classification. Unlike the original approach in Algorithm 1 that fixed k to cover 50% of pixels—for instance, setting k to 392 for 28x28 images—SMOOT adapts k based on each image's optimal masking percentage for maximizing accuracy. In SMOOT, described in Algorithm 2, k becomes a vector K_i, with $K_i(X)$ representing the percentage of pixels masked for image X in epoch i. The goal is to optimize model parameters for each image, adjusting K_i to improve accuracy. Initially, all K_i values are set to 50%. Adjustments are made per image, based on whether increasing or decreasing the percentage of masked pixels enhances accuracy. Images are categorized into two classes: Class I, where accuracy monotonically decreases with additional masking, and Class II, where initial masking increases accuracy before a subsequent decline. For Class I images, adjustments aim to reduce the masking percentage towards a minimum threshold, as depicted in Figure 1, maintaining minimal masking. In contrast, for Class II images, the adjustment seeks the accuracy peak, moving either towards more or less masking until the optimal point (highlighted with a blue dot in Figure 1) is reached. This adaptive approach allows for the nuanced optimization of feature masking, tailoring the process to individual image

characteristics and thereby enhancing model performance and feature relevancy in classification tasks.

Algorithm 2 SMOOT: Saliency Guided Mask Optimized Online Training

Training samples X, learning rate τ, hyperparameters λ and α, controls increase or decrease number of masking μ, Initialize f_θ {Preload or randomize for new training} Initialize K {50% to be consistent with prior work}
for $i = 1$ ***to*** *epochs* **do**
  for *minibatch* **do**
    {Get sorted index I for the gradient of output w.r.t the input.}
    1. $I = \text{sort}(\nabla_X f_{\theta_i}(X))$
    {compute $\widetilde{X}$ as the image with bottom k features of the original input masked.}
    2. $\widetilde{X} = M(i, K, I, X)$
    {Compute difference in softmax outputs when $softmax_i$ is ith highest softmax output}
    3. $\delta_1 = soft_1(\widetilde{X}) - soft_1(X)$
    4. $\delta_2 = \frac{1}{n-1}\sum_{i=2}^{i=n}(soft_i(\widetilde{X}) - soft_i(X))$
    5. $\delta = \alpha\delta_1 + (1-\alpha)\delta_2$
    {Find number of masking for next epoch}
    6. $K_{i+1}(X) = \max(K_{\min}, \min(K_{\max}, K_i + \lfloor \mu\delta \rfloor))$
    {Compute the loss function}
    7. $L_i = \mathcal{L}(f_{\theta_i}(X), y) + \lambda\mathcal{D}_{\mathcal{KL}}(f_{\theta_i}(X) \| f_{\theta_i}(\widetilde{X}))$
    {Use the gradient to update network parameters}
    8. $f_{\theta_{i+1}} = f_{\theta_i} - \tau\nabla_{\theta_i} L_i$
  end
end

The softmax output of X at epoch i is denoted as $softmax_i(X)$, and the masked version of X is represented as $\widetilde{X}$.

$$\delta_1 = softmax_1(\widetilde{X}) - softmax_1(X) \tag{1}$$

and the change in the top 2 to top n is computed using

$$\delta_2 = \frac{1}{n-1}\sum_{i=2}^{i=n}(softmax_i(\widetilde{X}) - softmax_i(X)) \tag{2}$$

In this equation, for top 5 accuracy, n should be equal to 5. We then use a weighted representation of change in softmax value using the equation:

$$\delta = \alpha\delta_1 + (1-\alpha)\delta_2 \tag{3}$$

For the generated results, in the result section of this paper, we have used the $n = 5$ and $\alpha = 0.7$, placing more priority on improvement in top 1 accuracy.

$$K_{i+1}(X) = \max(K_{\min}, \min(K_{\max}, K_i + \lfloor \mu\delta \rfloor)) \tag{4}$$

In this equation, K_{min} and K_{max} are the min and max percentages allowed for masking. In our experiment $K_{min} = 20$ and $K_{max} = 80$. Finally, the weight "μ" is a hyperparameter that determines the speed at which the masking percentage changes. The loss of the model is computed similarly to the previous SGT using:

$$L_i = \mathcal{L}(f_{\theta_i}(X), y) + \lambda\mathcal{D}_{\mathcal{KL}}(f_{\theta_i}(X) \| f_{\theta_i}(\widetilde{X})) \tag{5}$$

In which $\mathcal{L}$ is cross entropy loss and $\mathcal{D}_{\mathcal{KL}}$ is KL divergence.

$$\mathcal{D}_{\mathcal{KL}}(f_{\theta_i}(X) \| f_{\theta_i}(\widetilde{X})) = \sum_{x \in X} f_{\theta_i}(X) log(\frac{f_{\theta_i}(\widetilde{X})}{f_{\theta_i}(X)}) \tag{6}$$

The KL divergence is computed based on the similarity of $f_{\theta_i}(X)$ to $f_{\theta_i}(\widetilde{X})$. Using this updated loss, the gradients are then used to update the network parameters as follows:

$$f_{\theta_{i+1}} = f_{\theta_i} - \tau\nabla_{\theta_i} L_i \tag{7}$$

Our proposed solution's algorithm, outlined in Algorithm 2, takes input parameters such as training samples (X), initial feature masking (k), learning rate (τ), and hyperparameter (λ). It initializes model parameters (f_θ) and iterates through epochs and mini-batches. For each iteration, it finds the sorted index I of the gradient of output with respect to input ($\nabla_X f_{\theta_i}(X)$). Then, it generates a masked image from the input using a masking function ($M(\cdot)$), removing the lowest $K(i)$ features based on sorting vector I to generate $\widetilde{X}$. It adjusts the masking percentage based on the accuracy contrast between input and masked input. The loss function, computed in line 7, includes a weighted KL divergence comparing model output with original and masked inputs, incorporating saliency-guided regularization. Finally, it updates network parameters using gradient descent.

III. Experiments and Results

In our study, we assess the SMOOT method against traditional and SGT approaches by retraining models on MNIST [6], Fashion MNIST [22], CIFAR-10 [21], CIFAR-100 [21], and Caltech 101 [7] datasets.

A. Model Architecture

To replicate and enhance the results of the original SGT study by Ismail et al. [18], we employed various models across different datasets. Specifically, we utilized:

- **MNIST and Fashion-MNIST** datasets, a two-layer Convolutional Neural Network (CNN) with a kernel size of 3×3 and a stride of 1, followed by two fully connected layers. Dropout layers with rates of 0.25 and 0.5 were integrated for regularization. The hyperparameter α was set to 0.95, emphasizing the label's importance due to the datasets' lower complexity.
- **Caltech 101** dataset, we adopted a pre-trained *ResNet18* architecture, adding a 101-neuron output layer to accommodate the dataset's classification needs. The model was initially trained on the ImageNet dataset.
- **CIFAR datasets** were approached with the Tiny Transformer configuration, following the original 'deit' architecture with dimensions ($L = 12, d = 192, h = 3$), and included a 10-neuron classifier for the final layer.

The training was conducted on a single NVIDIA A100 GPU for 100 epochs, with a batch size of 256 for the MNIST and CIFAR datasets, and 128 for Caltech 101 and the Adadelta for optimization algorithm.

Table I summarizes the model architectures and hyperparameters used in our experiments:

Dataset	Model	Init K	τ	α	λ
MNIST	CNN	392	1	95%	1
Fashion-MNIST	CNN	392	1	95%	1
Caltech 101	Resnet 18	2500	1E-3	80%	1
CIFAR10	Trans.	512	1E-3	80%	1
CIFAR100	Trans.	512	1E-3	80%	1

TABLE I
Model Architecture and Hyperparameter Summary

B. Saliency Guided Training for Images

In the context of image classification using saliency, it is common to encounter redundant features that are not crucial for the model's prediction. Take the example of an object's background in an image, which occupies a significant portion but typically holds little relevance to the classification task. When the model's attention is directed toward the object itself, it is desirable for the background gradient (representing most of the features) to be close to zero, indicating its diminished importance. Figure 2 illustrates a comparison between the saliency map generated using our approach, the SGT in [18], and Traditional training (no saliency-guided training). Figure 2 provides this comparison for images selected from the MNIST dataset and Fashion MNIST dataset.

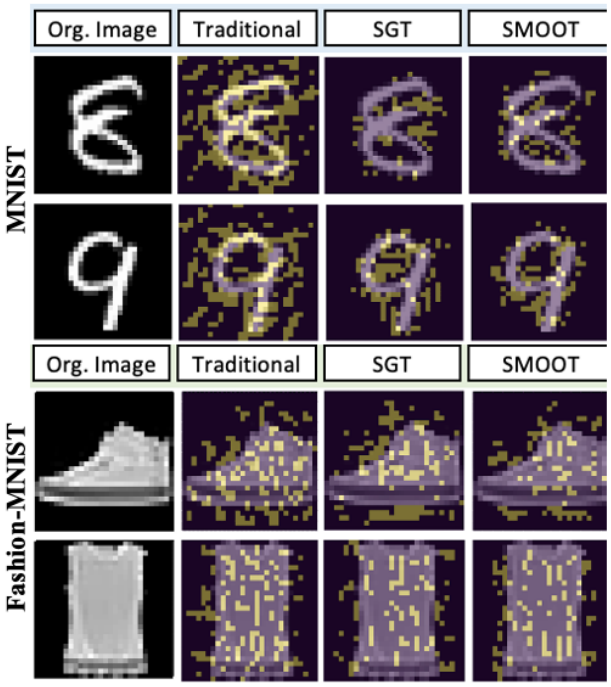

Fig. 2. The saliency map generated through the approach outlined in Ismail et al. [18] was applied to traditional, SGT, and SMOOT training methods. Visual representations for selected samples from the MNIST and Fashion-MNIST datasets are included. The results indicate that the saliency map produced by the SMOOT method aligns more closely with the target object for classification and exhibits a reduced number of erroneously identified salient pixels external to the object. Consequently, this evidence supports the conclusion that SMOOT's saliency maps are more interpretable for human analysis.

C. Model Accuracy Drop

In our study, we compare SMOOT with established methods like SGT [18]. Using various saliency techniques [14], we assess how feature ranking and elimination impact model accuracy. Our experiments on MNIST and Fashion MNIST datasets reveal that SMOOT induces a significant accuracy decline compared to traditional and SGT methods [14], [18]. This suggests SMOOT effectively removes less informative features, enhancing model performance. However, its effectiveness may vary in datasets with complex backgrounds. The

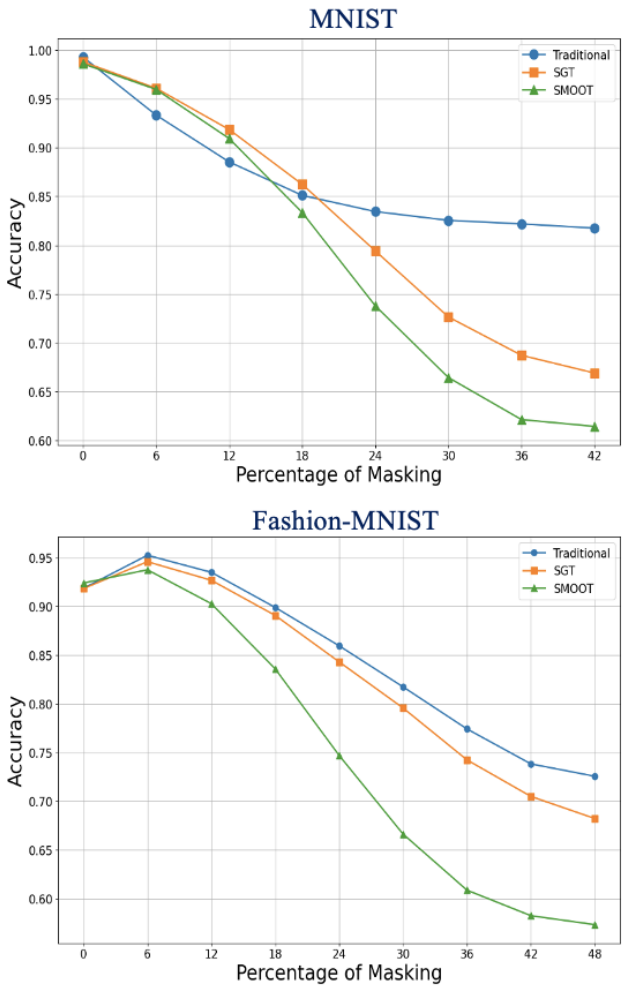

Fig. 3. Comparison of accuracy degradation in trained models upon masking high gradient inputs is conducted across three training approaches: standard cross-entropy-based training, SGT [18], and our proposed method, SMOOT. Enhanced saliency maps aid in reducing the misidentification of high gradient inputs, aligning better with the target object. Models with superior saliency maps exhibit sharper accuracy drops when high gradient pixels are removed. Notably, SGT shows a more rapid decline compared to traditional methods, while SMOOT surpasses SGT in the rate of accuracy decline, highlighting its superior ability to generate enhanced saliency maps.

results, detailed in tables II and III, show the Area Under the accuracy drop Curve (AUC) and accuracy metrics for different training methods on the MNIST and Fashion-MNIST datasets. A lower AUC value indicates better performance, representing a more substantial accuracy decline from eliminating non-informative features. SMOOT consistently outperforms both traditional and SGT methods, demonstrating enhanced accuracy and saliency in identifying and focusing on relevant features, which underscores its potential as a superior training methodology in specific dataset contexts. Our results in Figure 3 for the MNIST and Fashion-MNIST datasets reveal that the SMOOT model shows a greater drop in accuracy with increased masking than traditional and SGT models, highlighting its superior saliency in feature discernment.

MNIST	Min(K)	Med(K)	Max(K)	Acc(%)	AUC
Traditional	0	0	0	99.40	36.35
SGT	392	392	392	99.35	34.67
SMOOT	234	388	544	**99.40**	**33.16**

TABLE II

PERFORMANCE OF TRADITIONAL, SGT, AND SMOOT TRAINING BASED ON ACCURACY AND THE AREA UNDER THE ACCURACY CURVE (AUC) ON **MNIST** DATASET. A LOWER AUC SIGNIFIES SUPERIOR SALIENCY PERFORMANCE. WHILE SGT MAINTAINS A CONSISTENT 50% TOKEN DROP, SMOOT ADJUSTS THE VALUE OF K DYNAMICALLY. THE TABLE PRESENTS THE MIN, MAX, AND MEDIAN VALUES OF k FOR SMOOT.

Fashion	Min(K)	Med(K)	Max(K)	Acc(%)	AUC
Traditional	0	0	0	93.60	40.79
SGT	392	392	392	93.35	39.91
SMOOT	223	372	576	**93.65**	**36.18**

TABLE III

ACCURACY AND THE AREA UNDER THE ACCURACY CURVE (AUC) ON **FASHION-MNIST** DATASET. A LOWER AUC SIGNIFIES SUPERIOR SALIENCY PERFORMANCE.

D. Deep CNN

For deep CNN evaluation, we used the ResNet18 architecture to evaluate deep CNNs, showing in Table IV that SMOOT outperforms traditional and SGT models in accuracy on the Caltech 101 dataset. This highlights SMOOT's effectiveness. Figure 4 illustrates comparative saliency maps and gradient box plots, indicating SMOOT's precision in identifying salient features and minimizing irrelevant ones. The plots also show that SMOOT enhances gradients of key features, making it superior in producing accurate saliency maps compared to SGT and traditional methods.

Caltech	Min(K)	Med(K)	Max(K)	Acc(%)
Traditional	0	0	0	94.15
SGT	25000	25000	25000	94.50
SMOOT	10000	236885	28455	**95.10**

TABLE IV

COMPARING THE PERFORMANCE OF TRADITIONAL, SGT, AND SMOOT TRAINING BASED ON ACCURACY OVER **CALTECH 101** DATASET.

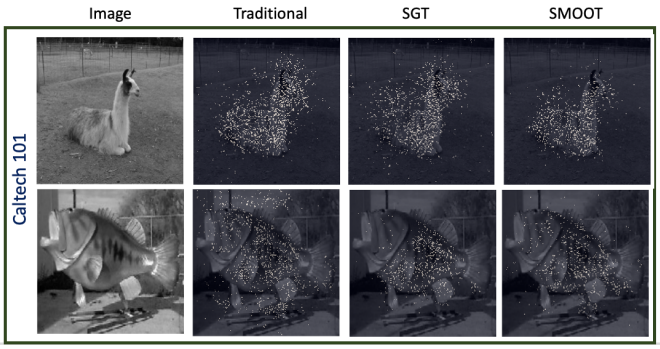

Fig. 4. Comparing SMOOT, SGT, and Traditional training methods for generating saliency maps from the Caltech 101 dataset using a Resnet 18 model. The best model uses high gradients to emphasize prominent pixels. A wider range in the box plot or distribution plot indicates more active pixels, with SMOOT showing superior results due to its mask considerations during training. The data from the saliency maps further supports the conclusion that SMOOT's saliency maps are more intuitively interpretable for humans.

E. SMOOT for Transformers

The Transformer model, originally from NLP and cited in Vaswani et al. [20], has significantly impacted deep learning, extending its success to computer vision with its self-attention mechanism effectively capturing long-range dependencies. Performance comparisons on CIFAR-10 and CIFAR-100, shown in Tables V and VI, demonstrate that SMOOT outperforms SGT and traditional training, enhancing accuracy through optimized masking and the SmoothGrad method.

Visual comparisons in Figure 5 between traditional, SGT, and SMOOT methods on input images highlight SMOOT's superiority, showcasing a taller gradient box and broader distribution. This indicates a more refined gradient adjustment, leading to not only higher model accuracy but also improved saliency map quality, making SMOOT a more effective and explainable model approach in computer vision.

Furthermore, Figure 5 provides a visual comparison among traditional, SGT, and SMOOT on a selected set of images. through the application of saliency maps to images.

CIFAR10	Min(K)	Median(K)	Max(K)	Accuracy
Traditional	0	0	0	95.65%
SGT	512	512	512	96.05%
SMOOT	204	488	753	**96.35%**

TABLE V

CIFAR10: THE ACCURACY COMPARES OUR MODEL (SMOOT) WITH A SALIENCY-GUIDED TRAINING (SGT) AND TRADITIONAL MODEL BY USING A TRANSFORMER.

CIFAR100	Min(K)	Median(K)	Max(K)	Accuracy
Traditional	0	0	0	75.75%
SGT	512	512	512	78.10%
SMOOT	362	432	682	**79.65%**

TABLE VI

CIFAR100: THE ACCURACY COMPARES OUR MODEL(SMOOT) WITH A SALIENCY-GUIDED TRAINING(SGT) AND TRADITIONAL MODEL BY USING TRANSFORMER.

IV. CONCLUSION

In the framework of SGT, the hyperparameter k is of paramount importance as it represents the "number of masking." This hyperparameter is instrumental in the learning process by pinpointing the most relevant pixels in each image. Also, such identification proves vital for improving accuracy, especially in datasets with larger image dimensions. Therefore, careful optimization of k is crucial to ascertain its optimal value. In this study, we present a novel approach to refine the selection k by evaluating the influence of masking pixels with low saliency scores on the accuracy of individual images within the dataset. We use MNIST and Fashion MNIST datasets with a simple CNN, Caltech 101 with a ResNet model, and CIFAR-10 with a transformer model. Our method improves model accuracy and interpretability compared to previous approaches, leading to better generalization and interoperability.

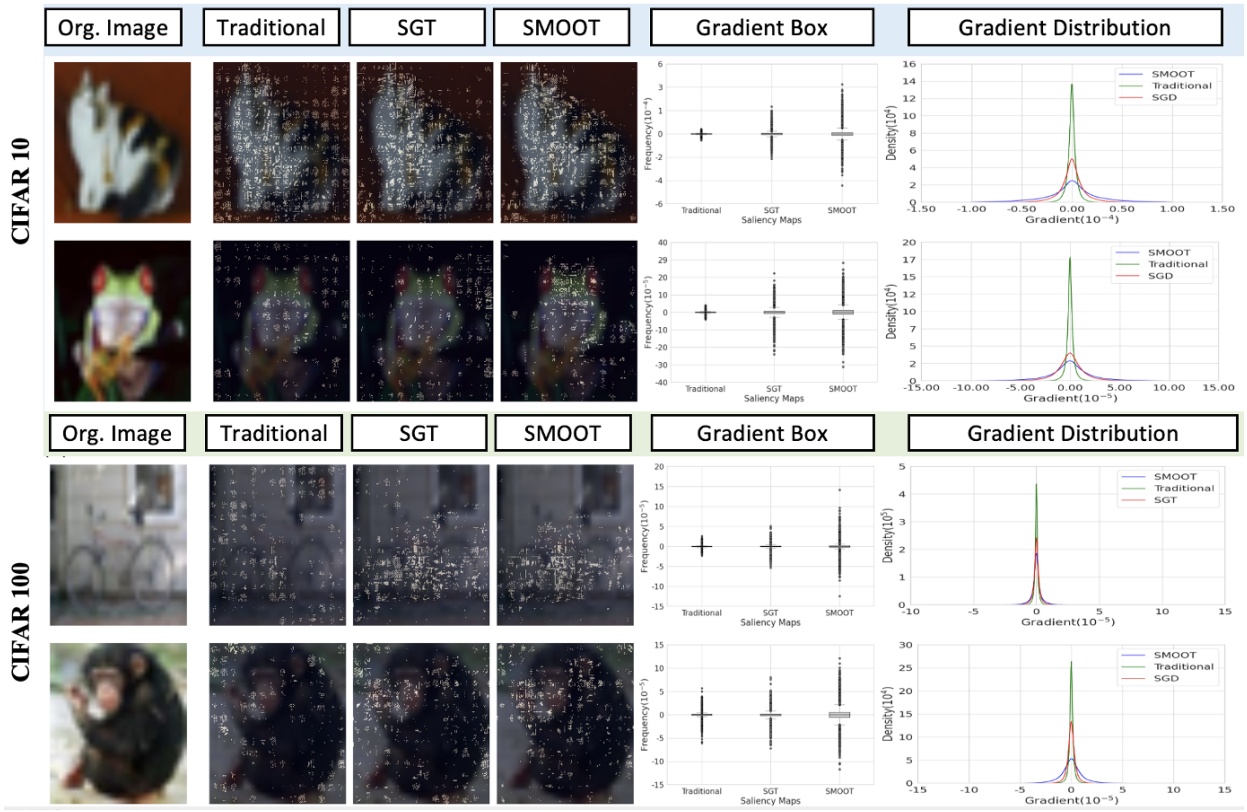

Fig. 5. Comparing SMOOT, SGT, and Traditional training with a Tiny transformer on CIFAR10 and CIFAR100 datasets, the best model highlights high gradients for prominent pixels. SMOOT, factoring in mask count during training, outperforms others, seen in wider ranges of active pixels in plots. Furthermore, the insights gained from the saliency maps reinforce the conclusion that SMOOT's saliency maps offer a more intuitive interpretation

REFERENCES

[1] Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M. & Elhadad, N. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. *Proceedings Of The 21th ACM SIGKDD International Conference On Knowledge Discovery And Data Mining*. pp. 1721-1730 (2015)

[2] Hooker, S., Erhan, D., Kindermans, P. & Kim, B. A benchmark for interpretability methods in deep neural networks. *Advances In Neural Information Processing Systems*. **32** (2019)

[3] Ghorbani, A., Abid, A. & Zou, J. Interpretation of neural networks is fragile. *Proceedings Of The AAAI Conference On Artificial Intelligence*. **33**, 3681-3688 (2019)

[4] Selvaraju, R., Cogswell, M., Das, A., Vedantam, R., Parikh, D. & Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. *Proceedings Of The IEEE International Conference On Computer Vision*. pp. 618-626 (2017)

[5] DeVries, T. & Taylor, G. Improved regularization of convolutional neural networks with cutout. *ArXiv Preprint ArXiv:1708.04552*. (2017)

[6] LeCun, Y., Cortes, C. & Burges, C. Mnist handwritten digit database. ATT Labs. (2010)

[7] Fei-Fei, L., Fergus, R. & Perona, P. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *2004 Conference On Computer Vision And Pattern Recognition Workshop*. pp. 178-178 (2004)

[8] Frosst, N. & Hinton, G. Distilling a neural network into a soft decision tree. *ArXiv Preprint ArXiv:1711.09784*. (2017)

[9] Ghaeini, R., Fern, X., Shahbazi, H. & Tadepalli, P. Saliency learning: Teaching the model where to pay attention. *ArXiv Preprint ArXiv:1902.08649*. (2019)

[10] Ismail, A., Gunady, M., Corrada Bravo, H. & Feizi, S. Benchmarking deep learning interpretability in time series predictions. *Advances In Neural Information Processing Systems*. **33** pp. 6441-6452 (2020)

[11] Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M. & Kim, B. Sanity checks for saliency maps. *Advances In Neural Information Processing Systems*. **31** (2018)

[12] Ancona, M., Ceolini, E., Öztireli, C. & Gross, M. Towards better understanding of gradient-based attribution methods for deep neural networks. *ArXiv Preprint ArXiv:1711.06104*. (2017)

[13] Singh, K. & Lee, Y. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. *2017 IEEE International Conference On Computer Vision (ICCV)*. pp. 3544-3553 (2017)

[14] Kindermans, P., Schütt, K., Müller, K. & Dähne, S. Investigating the influence of noise and distractors on the interpretation of neural networks. *ArXiv Preprint ArXiv:1611.07270*. (2016)

[15] Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K. & Samek, W. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS One*. **10**, e0130140 (2015)

[16] Perkins, S., Lacker, K. & Theiler, J. Grafting: Fast, incremental feature selection by gradient descent in function space. *The Journal Of Machine Learning Research*. **3** pp. 1333-1356 (2003)

[17] Singla, S., Wallace, E., Feng, S. & Feizi, S. Understanding impacts of high-order loss approximations and features in deep learning interpretation. *International Conference On Machine Learning*. pp. 5848-5856 (2019)

[18] Ismail, A., Corrada Bravo, H. & Feizi, S. Improving deep learning interpretability by saliency guided training. *Advances In Neural Information Processing Systems*. **34** pp. 26726-26739 (2021)

[19] Wang, L., Wu, Z., Karanam, S., Peng, K., Singh, R., Liu, B. & Metaxas, D. Sharpen focus: Learning with attention separability and consistency. *Proceedings Of The IEEE/CVF International Conference On Computer Vision*. pp. 512-521 (2019)

[20] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, Ł. & Polosukhin, I. Attention is all you need. *Advances In Neural Information Processing Systems*. **30** (2017)

[21] Krizhevsky, A., Hinton, G. & Others Learning multiple layers of features from tiny images. (Toronto, ON, Canada,2009)

[22] Xiao, H., Rasul, K. & Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *ArXiv Preprint ArXiv:1708.07747*. (2017)