OPUS-HSO

Repositorium der Hochschule Offenburg
https://opus.hs-offenburg.de

HOCH SCHULE OFFEN BURG

Reliable Evaluation of Attribution Maps in CNNs: A Perturbation-Based Approach

Lars Nieradzik, Henrike Stephani, Janis Keuper

Zitiervorschlag im APA Stil:

Nieradzik, L., Stephani, H., & Keuper, J. (2025). Reliable Evaluation of Attribution Maps in CNNs: A Perturbation-Based Approach. *International Journal of Computer Vision*, *133*(5), 2392–2409. https://doi.org/10.1007/s11263-024-02282-6

Abstract

In this paper, we present an approach for evaluating attribution maps, which play a central role in interpreting the predictions of convolutional neural networks (CNNs). We show that the widely used insertion/deletion metrics are susceptible to distribution shifts that affect the reliability of the ranking. Our method proposes to replace pixel modifications with adversarial perturbations, which provides a more robust evaluation framework. By using smoothness and monotonicity measures, we illustrate the effectiveness of our approach in correcting distribution shifts. In addition, we conduct the most comprehensive quantitative and qualitative assessment of attribution maps to date. Introducing baseline attribution maps as sanity checks, we find that our metric is the only contender to pass all checks. Using Kendall's τ rank correlation coefficient, we show the increased consistency of our metric across 15 dataset-architecture combinations. Of the 16 attribution maps tested, our results clearly show SmoothGrad to be the best map currently available. This research makes an important contribution to the development of attribution maps by providing a reliable and consistent evaluation framework. To ensure reproducibility, we will provide the code along with our results.

Nutzungsbedingungen

Dieses Dokument wird unter diesen Bedingungen zur Verfügung gestellt:
Creative Commons - CC BY - Namensnennung 4.0 International
Für weitere Informationen siehe:
https://creativecommons.org/licenses/by/4.0/deed.de

Kontakt

Hochschule Offenburg | Bibliothek
Badstraße 24
77652 Offenburg
Telefon: (0781) 205-240
E-Mail: bibliothek@hs-offenburg.de
www.hs-offenburg.de/bibliothek

International Journal of Computer Vision (2025) 133:2392–2409
https://doi.org/10.1007/s11263-024-02282-6

Reliable Evaluation of Attribution Maps in CNNs: A Perturbation-Based Approach

Lars Nieradzik[1] · Henrike Stephani[1] · Janis Keuper[2]

Received: 8 September 2023 / Accepted: 20 October 2024 / Published online: 23 November 2024
© The Author(s) 2024

Abstract

In this paper, we present an approach for evaluating attribution maps, which play a central role in interpreting the predictions of convolutional neural networks (CNNs). We show that the widely used insertion/deletion metrics are susceptible to distribution shifts that affect the reliability of the ranking. Our method proposes to replace pixel modifications with adversarial perturbations, which provides a more robust evaluation framework. By using smoothness and monotonicity measures, we illustrate the effectiveness of our approach in correcting distribution shifts. In addition, we conduct the most comprehensive quantitative and qualitative assessment of attribution maps to date. Introducing baseline attribution maps as sanity checks, we find that our metric is the only contender to pass all checks. Using Kendall's τ rank correlation coefficient, we show the increased consistency of our metric across 15 dataset-architecture combinations. Of the 16 attribution maps tested, our results clearly show SmoothGrad to be the best map currently available. This research makes an important contribution to the development of attribution maps by providing a reliable and consistent evaluation framework. To ensure reproducibility, we will provide the code along with our results.

Keywords Explainability · Saliency maps · Class activation maps · Attribution maps · Adversarial attacks

1 Introduction

The explainability of learned models is a crucial, yet mostly unachieved property towards a general utilization and acceptance of machine learning techniques (Gilpin et al., 2018). In order to succeed, many practical applications require solutions that provide not only high test accuracy, but also robustness, uncertainty estimates for predictions, and ways to make decision processes understandable for humans. Although recent deep learning methods have been very successful in achieving great advances in terms of model accuracy on many different tasks (Zarándy et al., 2015), the other properties are mostly still the subject of increasing research efforts.

A current standard tool for the visualization of decision processes in neural networks are so-called *Attribution Methods* (Zhang et al., 2021). In the case of *Computer Vision* tasks with image input data, attribution methods are used to compute *Attribution Maps* (AM) (also known as *Saliency Maps* (Zhang et al., 2021)) of individual input images on a given trained model. AMs provide a human-interpretable visualization of the image regions' weighted impact on the model, allowing intuitive explanations of the complex internal mappings of model predictions. However, the qualitative nature of AMs makes it very hard to validate their actual correctness or to define a performance measure for AM methods (Nieradzik et al., 2024). Fig. 1 gives a representative example of the large variations in the visual results of four different prominent AM algorithms, intuitively showing the need for quantitative AM evaluation methods in terms of the following, practical questions: **(i) How can we evaluate the correctness of AM outputs? (ii) How can we compare the performance of the many recently published AM algorithms? (iii) Which**

Communicated by Jianfei Cai.

✉ Lars Nieradzik
lars.nieradzik@itwm.fraunhofer.de

Henrike Stephani
henrike.stephani@itwm.fraunhofer.de

Janis Keuper
keuper@imla.ai

[1] Image Processing, Fraunhofer ITWM, Fraunhofer Platz 1, Kaiserslautern 67663, Germany

[2] Institute for Machine Learning and Analysis (IMLA), Offenburg University, Badstr. 24, Offenburg 77652, Germany

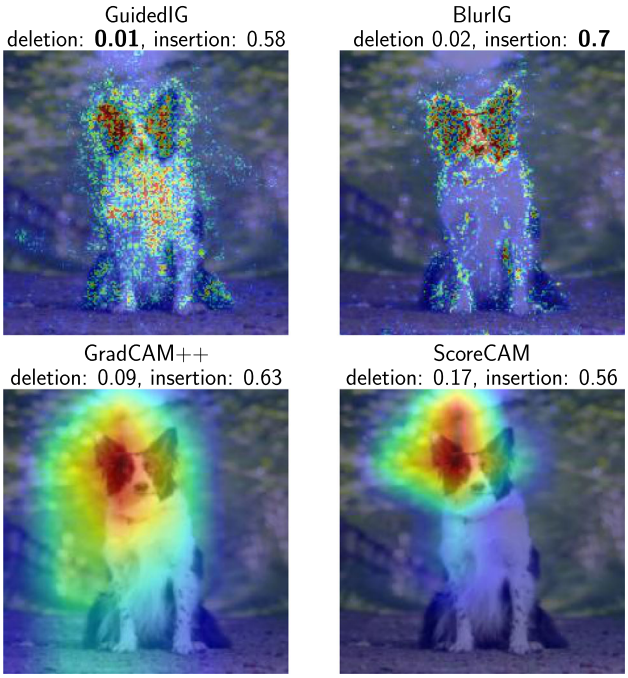


Fig. 1 Visualization of four different attribution maps (AM) and two evaluation methods on the same input image and the same model (EfficientNet-B0 (Tan & Le, 2019)). Visually, it is impossible to objectively determine which result gives the "best" estimate of the image regions with the highest impact on the model decision. Current evaluation methods of the used AMs also give conflicting results: the *Deletion* method (lower is better) (Fong & Vedaldi, 2017; Petsiuk et al., 2018) ranks *Guided Integrated Gradients (IG)* (Kapishnikov et al., 2021) first, while the *Insertion* method (higher is better) points towards *Blur IG* (Xu et al., 2020). Refer also to Fig. 2 for a comparison between more images

attribution method should we use for the investigation of a given research question or development goal?

To address these questions, we propose a novel quantitative evaluation method for AMs in *Convolutional Neural Networks* (CNN) (LeCun et al., xxxx), utilizing adversarial perturbations to verify the given importance of image regions to CNN predictions by attribution methods. The key contributions of this paper are:

- We present a novel approach for attribution map evaluation that addresses the distributional shift problem encountered in widely used insertion/deletion metrics. Our method uses adversarial perturbations instead of pixel modifications, resulting in a more robust evaluation framework.
- We perform the most comprehensive quantitative and qualitative evaluation of attribution maps to date, covering 16 attribution methods across 15 dataset-architecture combinations.
- We demonstrate the increased consistency and reliability of our metric using Kendall's rank correlation coefficient τ, baseline attribution maps, smoothness, monotonicity and a qualitative analysis.

2 Related Work

2.1 Attribution Methods

Since the literature on attribution methods has become extensive in recent years, and many approaches are based on earlier methods, we retreat to the discussion and evaluation of some of the most widely used AM techniques in the context of CNNs.

Full back-propagation methods, like *Gradients* (Simonyan et al., 2014), have been the basis for the first AM approaches for classification models: they compute a gradient on a learned neural network f with respect to a given input X for a fixed target class c: $\frac{\partial f^{(c)}}{\partial X}$. The derivation of this approach becomes quite intuitive, when we retreat to a simple linear decision function $f(x) = w^T x$. Then, the weights w are a direct indicator of the input importance, as we have $\frac{\partial f}{\partial x} = w$. The later introduced *DeConvNet* (Zeiler & Fergus, 2013) and *Guided backpropagation* (Springenberg et al., 2015) modify the gradients of the widely used *ReLU* activation to enable the backward flow of negative gradients. Another extension of *Gradients* has been introduced by *SmoothGrad* (Smilkov et al., 2017), which computes the gradient multiple times by adding Gaussian noise to the input and averaging the results.

Path backpropagation methods parameterize a path from a fixed baseline image to the given input image and compute derivatives analogous to *Gradients* along this path. For example, *Integrated Gradients* (Sundararajan et al., 2017) parameterizes the path as straight line $\gamma = (\gamma_1, \ldots, \gamma_n) : [0, 1] \to \mathbb{R}^n$ and $\gamma(\alpha) = X' + \alpha(X - X')$ where $X' \in \mathbb{R}^n$ is a black (all zero) image, $X \in \mathbb{R}^n$ is the test input and $\alpha \in [0, 1]$. Then the line integral is

$$IG_i(X) := \int_{\alpha=0}^{1} \frac{\partial f^{(c)}(\gamma(\alpha))}{\partial \gamma_i(a)} \frac{\partial \gamma_i(\alpha)}{\partial \alpha} d\alpha$$

with $f^{(c)} : \mathbb{R}^n \to [0, 1]$ being the neural network. In practice, the integral is approximated by the Riemann sum. *Blur Integrated Gradients* (Xu et al., 2020) proposed to replace the black baseline by filtering the image with the Gaussian blur filter. *Guided Integrated Gradients* (Kapishnikov et al., 2021) enhanced the straight line path by an adaptive path that depends on the model.

Class Activation Maps (CAM) can be considered as variants of *Gradients* in most cases. Instead of computing the gradient with respect to the input X, the back-propagation step is stopped at a chosen layer. For example, the *attribution map* for the popular *GradCAM* (Selvaraju et al., 2019)

is given by

$$GradCam(X) := ReLU\left(\sum_k w_k^{(c)} A_k\right),$$

with $w_k^{(c)} = \frac{1}{H \cdot W} \sum_{i=1}^{H} \sum_{j=1}^{W} \frac{\partial f^{(c)}}{\partial A_k(i,j)}$ and where H is the image height, W the image width, and A is the activation of the chosen layer. It is important to note, that due to the pyramid architecture of CNNs, the resolution of CAMs is typically much lower than the input images. Hence, CAMs need to be up-sampled, or inputs have to be down-sampled to compute an AM visualization. *GradCAM++* (Selvaraju et al., 2019) improved the original by weighting the gradients differently (Chattopadhay et al., 2018). *Smooth GradCAM++* (Omeiza et al., 2019) adds additional Gaussian noise to the input, like *SmoothGrad*. Finally, *LayerCAM* (Jiang et al., 2021) and *XGradCAM* (Fu et al., 2020) also provide a different weighting schemes for the computed gradients.

Unlike the previous methods, *ScoreCAM* (Wang et al., 2020) does not require the computation of gradients (and is therefore not a variant of Gradients). The importance of channels in an intermediate layer is given by the confidence change when parts of the activation values are removed.

There are countless other *attribution map* methods that are improvements, combinations and extensions of the aforementioned methods (Ancona et al., 2017) such as *SS-CAM* (Wang et al., 2020), *IS-CAM* (Naidu et al., 2020), *Ablation-CAM* (Desai & Ramaswamy, 2020), *FD-CAM* (Li et al., 2022), *Group-CAM* (Zhang et al., 2021), *Poly-CAM* (Englebert et al., 2022), *Zoom-CAM* (Shi et al., 2020), *Recipro-CAM* (Byun & Lee, 2023) and *EigenCAM* (Muhammad & Yeasin, 2020).

Black-box methods. Black box methods rely on masking the input in some way, e.g. randomly occluding the input image and recording the change in the class probabilities. A notable example is *RISE* (Petsiuk et al., 2018) and its precursor introduced in Zeiler and Fergus (2013). Other examples are Fong et al. (2019), Fong and Vedaldi (2017), Petsiuk et al. (2018), Ribeiro et al. (2016).

2.2 Evaluation of Attribution Methods

Given the large number of available attribution methods introduced in the previous section, users are left with the question which AM method to choose for their application (see Fig. 1). However, most publications that introduce novel AM methods are providing only qualitative visualizations of their results, showing selected input samples in direct comparison with other methods. Up to date, there is no commonly accepted benchmark that would provide a quantitative metric or ranking of *attribution map* algorithms for a given task. In the following, we review the few existing prior works. Formally, we define a score function for the evaluation of AMs as $g(h, f, X) = o$ for $o \in \mathbb{R}$ with attribution method h, CNN f and input X. For notational convenience, we leave out the superscript of the class c and only consider a single image.

***Average Drop* (AD) and *Increase in Confidence* (IIC).** *Average drop* has been suggested by Chattopadhay et al. (2018). It is defined as $g(h, f, X) := \max(f(X) - f(X \circ h(X)), 0)$, where $f(X)$ is the model's *softmax* output score (confidence) for class c. The operator $\circ$ denotes element-wise multiplication. The output $f(X \circ h(X))$ is the confidence of the model in class c with only the so-called *explanation map* as input. The *explanation map* is derived by multiplying the input with the *attribution* map, such that unimportant pixels are hidden.

Increase in confidence is similarly defined as $g(h, f, X) := \mathbf{1}[f(X) < f(X \circ h(X))]$ where $\mathbf{1}[\cdot]$ is the indicator function. Both methods quantify how much the output probabilities decrease or increase when given only the most salient regions in the input image.

***Complexity* (CP), *Coherency* (CH) and *ADCC*.** Poppi et al. (2021) suggested measuring the number of pixels that are highlighted in a *attribution map*. They define $g(h, f, X) := ||h(X)||_1$. The basic idea follows the argument that a good *attribution map* should highlight only a small area.

Coherency is another evaluation score by the same authors. It measures how much the original *attribution map* differs from the *attribution map* of the *explanation map*. It is defined by the correlation coefficient $g(h, f, X) := \frac{\mathrm{Cov}(h(X), h(X \circ h(X)))}{\sigma_h \sigma_{hh}}$, where $h(X)$ is the input image *attribution map* and $h(X \circ h(X))$ is the *attribution map* of the *explanation map*. Finally, the authors combine average drop, complexity and coherency to a single score by using the harmonic mean $\mathrm{ADCC} := 3\left(\frac{1}{\mathrm{CH}} + \frac{1}{1-\mathrm{CP}} + \frac{1}{1-\mathrm{AD}}\right)^{-1}$. CP and CH have several drawbacks as evaluation functions. An attribution map with a few highlighted pixels is not always optimal. For example, if we look at an empty image, CP would always say that this map is optimal. Similarly, it is easy to construct counterexamples for CH where the score is high but the AM is meaningless. Since ADCC is only a combination of these metrics, this function will suffer from the same limitations.

***Deletion* and *Insertion*.** Our approach for defining an evaluation function for saliency maps is based on these two metrics. They were introduced by Fong and Vedaldi (2017) and Petsiuk et al. (2018). These metrics will be expounded upon in detail in the next section. Del/Ins can be seen as generalizations of AD and IIC. AD and IIC evaluate the probability only once, while Del/Ins consider the input image as a sequence of images $I_1, I_2, \ldots, I_n$ and calculate the change in the probability of the whole modification process.

Other evaluation methods. *Sparsity* (Gomez et al., 2022) follows a similar idea as *Complexity*. The authors also sug-

gest using the correlation coefficient instead of AUC for the computation of the deletion and insertion scores.
Another approach is the *pointing game* (Zhang et al., 2016), with a variation introduced in Raatikainen and Rahtu (2022). The basic idea is to verify whether the pixel with the highest saliency score is contained in the bounding box of a specific object.

3 A Perturbation-Based Evaluation Method

First, let us establish the need for introducing an evaluation function. As we have visually demonstrated in Fig. 1, saliency maps can exhibit significant variations. To precisely quantify these differences, we evaluate the similarity between individual *attribution maps* (AM). To achieve this, we utilize the Pearson correlation coefficient, defined as:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}},$$

where x_i is the ith pixel in the AM and $\bar{x}$ is the sample mean. Similarly, y_i is the ith pixel of the second AM. The output range of r_{xy} is $[-1, 1]$, with 1 indicating the highest similarity and ≤ 0 a low similarity. $r_{xy} = 0$ is intuitively random noise and $r_{xy} = -1$ an "inverted" saliency map.

We assessed the similarity across a subset of 1000 images from ImageNet, employing both the ResNet-50 and ConvNeXt-tiny architectures. In Fig. 2, we present the results for ResNet-50, revealing an average similarity of 48%. Remarkably, for ConvNeXt-tiny, we observed an even lower similarity of 22%.

Given the substantial variation among these attribution maps, it becomes imperative to define a metric for evaluating which AMs most accurately represent what CNNs are truly "seeing".

Defining an evaluation function for *attribution maps* is inherently difficult because the actual ground truth is not known. Some publications (Jiang et al., 2021) on AM methods use segmentation masks from PASCAL VOC (Everingham et al., 2009) or COCO (Lin et al., 2014) as pseudo ground truth to evaluate their results. However, we strongly argue against this practice since it imposes an AM score based on which regions CNNs *should* base its decisions on, whereas the task of AMs is to show on which regions it actually does rely on. For example, if a model dominantly uses the image background for its predictions, AM should reveal this flaw instead of trying to segment the foreground objects.

More sophisticated approaches compensate the lack of ground truth information by use of surrogate functions. Treating the given neural network as a black box, the surrogate functions are used to manipulate the model input based on the AM under evaluation, deriving an AM score from the model response.

However, an underlying challenge emerges – current surrogate functions induce a distribution shift in the image due to significant pixel alterations. This shift renders evaluation scores unreliable, often marking subpar saliency maps as superior, depending on the amount of alterations. Our proposed evaluation score overcomes these limitations inherent in previous metrics by rectifying the distribution shift. We first introduce the most prominent of these surrogate functions: the Deletion and Insertion score functions based on which we develop our evaluation measure.

Deletion and Insertion. The approach introduced by Fong and Vedaldi (2017) and Petsiuk et al. (2018) revolves around the iterative deletion and insertion of pixels in input images. These pixels are chosen according to the highest values in the associated *attribution map*. The deletion method commences with the original input image, and subsequently proceeds by iteratively masking sequentially significant regions, step by step, with a value of 0. After each iteration step, we give the modified image to the neural network to generate a probability. Figure 3 visually illustrates the deletion process. In contrast, for the insertion method, the procedure is inverted. Instead of directly introducing pixels into an initially empty image, a common variant of the insertion method first applies Gaussian blur to the image. This results in a smoothed version of the initial image which then serves as the starting point for the iterative insertion of pixels.

The outcome of this procedure can be graphically represented by plotting the number of inserted or deleted pixels along the x-axis and the neural network probability of the target class along the y-axis. Fig. 4a shows an example for the Insertion metric. When the image is black ("Pixels changed" is 0.0), the probability of the target class is zero. When all the original pixels are in the image, the probability is approx. 65%.

The metric's evaluation is based on calculating the area under this plotted curve (AUC). For the "Deletion" metric, a good attribution map is indicated by a low AUC – as we assume that all the most relevant pixels (represented by the map) for the models decision process have been masked out. The faster the probability for the ground truth class diminishes to zero, the more effective the map is considered to be. Conversely, in the case of the "Insertion" metric, the aim is for the probability to return fast to its original value, as only those pixels deemed most relevant by the saliency map are used for the models decision. Consequently, a high AUC indicates a good saliency map for the "Insertion" metric. In the evaluation section, we call the methods InsBlur (insertion with blurring), Ins (insertion without blurring) and Del (deletion).

Adversarial perturbation. The original motivation behind the Deletion/Insertion metric was to hide/emphasize the

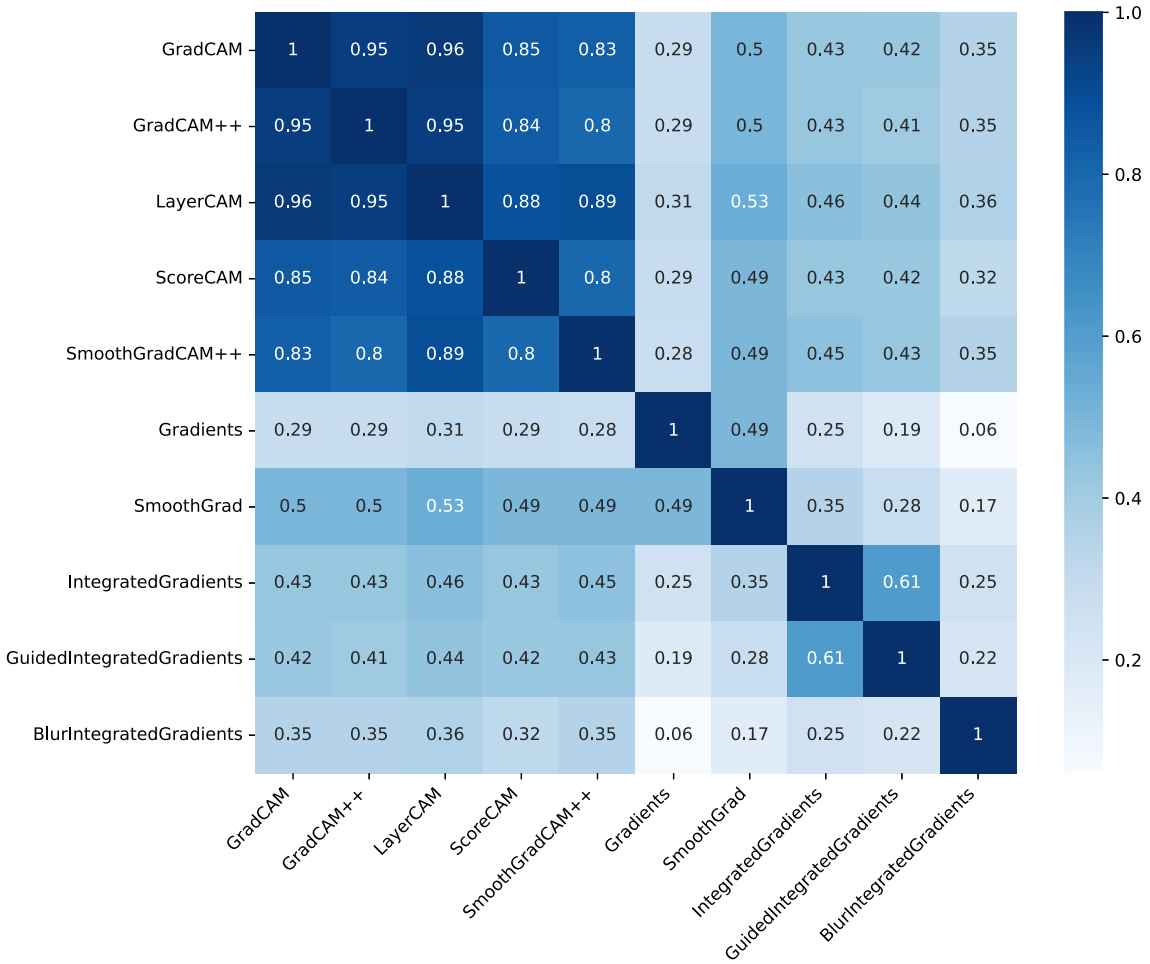


Fig. 2 This plot illustrates the degree of similarity among all attribution maps. The matrix was computed by averaging the individual correlation results across all attribution maps in the ImageNet dataset with ResNet-50

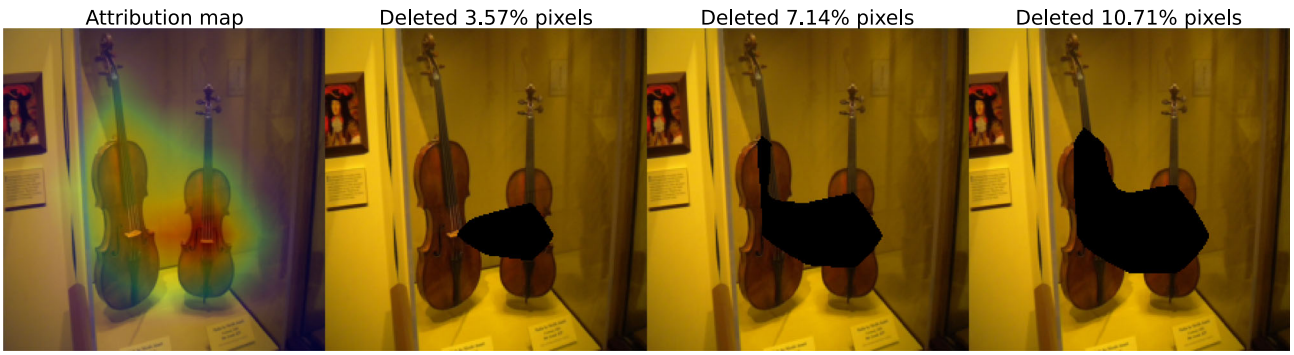
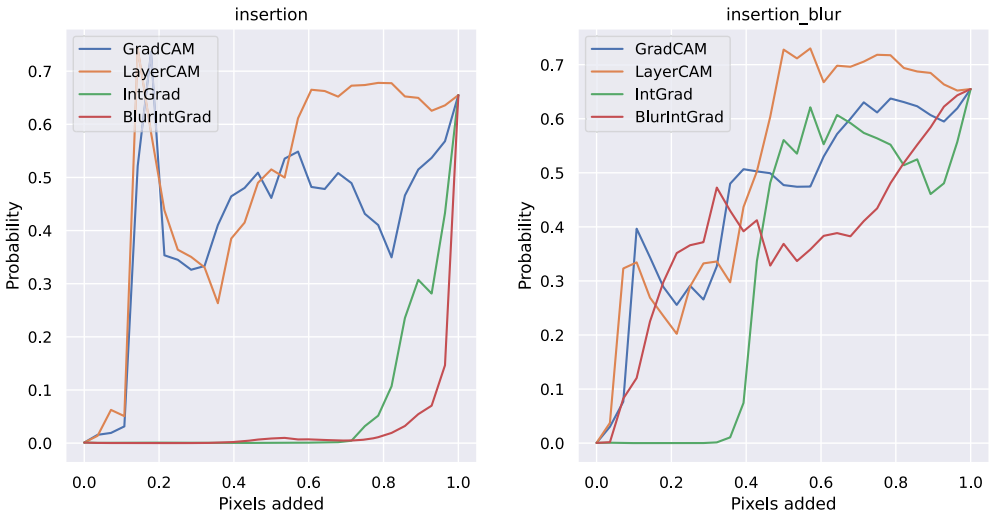
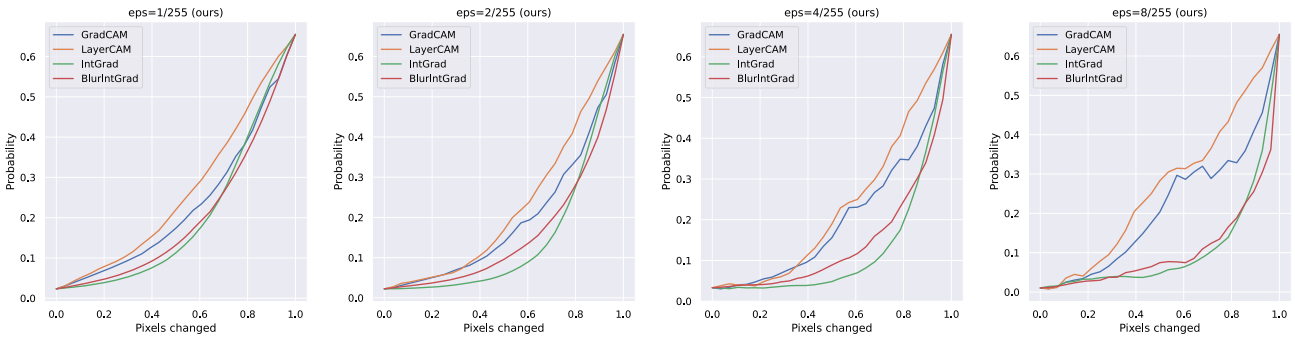


Fig. 3 The series of images showcases the process of the Deletion methods, applied to the original image. The first image combines the original image with the saliency map overlay. Subsequent images (2nd to 4th) depict three stages of deletion, with progressively less important regions being zeroed out, as indicated by the colors in the first image



(a) Insertion and Insertion Blur



(b) Our Perturbation method for Different Epsilon Values

Fig. 4 Insertion, Insertion Blur, and our Perturbation method applied to one image. **a** Shows the Insertion and Insertion Blur methods. **b** Displays our Perturbation method for different epsilon values. Probability refers to the neural network's confidence for the selected class. The order of removing the perturbed pixels **b** and inserting the pixels **a** depends on the saliency map. The highest values are changed first. AUC values are computed for each saliency map, condensing the outcome into a single scalar value. Refer to Table 5 for numerical results using more than one image

region responsible for determining the class. Each of the modified images is given to the CNN to produce a probability. When the probability of the ground truth class swiftly reaches zero/one, it signifies that the saliency map correctly identified the most crucial pixels for the original class decision. Nonetheless, an issue arises when setting pixels to zero: these types of images might not have been encountered during the neural network's training. Consequently, the CNN probability exhibits fluctuations (as seen in Fig. 4a. The pixels could be misconstrued by the neural network as representing an additional object. To circumvent this, we propose concealing the significant regions within the image through minimal alterations. This approach enhances the score's reliability by maintaining a distribution closer to the original one.

To achieve this, we develop a gradient-based method using the FGSM adversarial attack. An adversarial attack's primary objective is to modify the image X in the smallest possible way to a perturbed image X^* that leads to a change in the models classification decision (Akhtar & Mian, 2018). In other words, the class with the highest probability for the initial image X should differ from the class with the highest probability for the perturbed image X^*.

Adversarial attacks can be categorized in various ways. Some attacks utilize gradient information to determine the necessary perturbations, while others rely solely on the model's class decision (Rauber et al., 2017). Certain attacks do not even require knowledge of the true class label. Some approaches apply perturbations to all pixels, whereas others selectively target individual pixels for modification (Su et al., 2017).

To define our metric for attribution maps, we use a gradient-based attack. This choice is driven by the substantial

speed advantage of gradient-based attacks, as they maximize specific loss functions. It is noteworthy that our metric's effectiveness does not hinge on a perfect attack rate. Hence, stronger but slower attacks are not needed. We show in our ablation study that the results do not change when switching from the FGSM to the PGD attack.

Since we are concerned about distribution shifts, we apply perturbations uniformly to all pixels. This strategy introduces a milder distribution shift compared to selectively perturbing specific pixels by significant amounts. Reviewing the literature on detecting adversarial attacks, shows that certain stronger attacks such as the Carlini & Wagner attack (Carlini & Wagner, 2016) are easier to detect by some methods (Liang et al., 2021; Xu et al., 2017). Additionally, one-pixel attacks such as Su et al. (2017) cannot be used for defining a metric because for saliency maps potentially every pixel can be important. This suggests that small uniform perturbations are better suited for defining a metric.

Generating an adversarial example X^* with a gradient-based attack can be described by the following optimization problem

$$\arg\max_{\delta \in S} L(f(X+\delta), y), \tag{1}$$

where the perturbed samples are given by $X^* = X + \delta$, L is the loss function, y is the target class and $f(\cdot)$ is the neural network (Li et al., 2021). The set of allowed perturbations is $S \in \mathbb{R}^d$, usually chosen to be $S = \{\delta \big| ||\delta||_\infty \leq \epsilon\}$ for a small $\epsilon \in \mathbb{R}$. This ϵ is not necessarily an infinitesimal change as in mathematics. The value is usually chosen to be $\frac{k}{255}$, where $k \in \{1, 2, \ldots, 255\}$.

A solution to this gradient-based optimization problem using the ℓ_∞-norm for S is the *fast gradient sign attack* (FGSM) (Goodfellow et al., 2014), where $\delta = \epsilon \cdot \operatorname{sgn}\left(\frac{\partial L}{\partial X}\right)$ and $\operatorname{sgn}(\cdot)$ is the sign function.

To each pixel of the original image either $-\epsilon$ or ϵ is added. When a pixel value is below 0 or above 1, the pixels are clipped back to the original domain. This does not affect the norm, since the strength of the attack is determined by the maximum. Therefore, the image attack's strength after clipping is still ϵ.

It is important to note that adversarial attacks typically assume that images are normalized to the range [0, 1], which simplifies the optimization problem by working in a continuous space. However, we want to specifically consider the case of discrete 8-bit images with values in the range $\{0, 1, \ldots, 255\}$. In this context, the minimal perturbations that can be applied to an image are 1 and -1.

We opt for discrete images to remain as close as possible to the original image distribution. When saving an image as a JPG file, all values become byte-valued, without fractional components. The FGSM attack mentioned earlier aligns with this requirement because it introduces ± 1 perturbations for all pixels when $\epsilon = \frac{1}{255}$, satisfying the minimal perturbation criterion.

Other attacks do not necessarily satisfy the requirements of being discrete and uniform. For instance, the ℓ_2-norm variant of FGSM with $\delta = \epsilon \cdot \frac{\partial L}{\partial X} \left(\| \frac{\partial L}{\partial X} \|_2\right)^{-1}$ is neither uniform nor discrete.

Deletion/Insertion as ℓ_∞ adversarial attack. Interestingly, we can view the original Deletion and Insertion metrics as specific instances of an ℓ_∞ adversarial attack. Consider the FGSM attack $\delta = \epsilon \cdot \operatorname{sgn}\left(\frac{\partial L}{\partial X}\right)$ where $\frac{\partial L}{\partial X} < 0$ for all pixels of the image $X \in [0, 1]^{n \times m}$, and $\epsilon = \frac{255}{255}$. In this case, $\delta = -1$ and $X^* = \max(X + \delta, 0) = 0$. If the network interprets all input pixels as inversely correlated with the loss function, the entire image is blacked out. However, typically such high values ϵ are not employed, as adversarial attacks should not be perceptible to the human eye. Hence, if ϵ is a small value like $\frac{1}{255}$ or $\frac{8}{255}$, it would correspond to a slight increase or decrease in brightness.

These subtle brightness adjustments do not disrupt the output probability distribution (as shown in Fig. 4b). Moreover, color adjustments are a standard data augmentation technique during training. The only difference to the standard adversarial attack is that instead of changing the color globally, it is changed for each pixel individually. Hence, the attacks produce images, that are likely to have qualitatively occurred as training samples.

Defining the score function. We start with a fully perturbed image where to every pixel either 1 or -1 was added with the FGSM attack. Gradually, this perturbation is systematically undone using the AM, as exemplified in Fig. 4b. As the perturbation is progressively reversed, we return to the original probability. However, when the network converges more rapidly to the initial probability, it signifies that the saliency map accurately pinpointed the pertinent regions. Hence, a high AUC value is essential for a saliency map to be deemed effective.

Unlike Deletion and Insertion, our functions are smooth and monotonic. This is obvious by comparing Fig. 4a with Fig. 4b. Additional evidence is given in the evaluation section.

Since we only change the pixels by ± 1, it is the smallest conceivable attack with $\epsilon = \frac{1}{255}$. While it is possible to use higher ϵ values such as $\frac{8}{255}$ or $\frac{4}{255}$, this can also cause a distribution shift. We can see in Fig. 4b that higher epsilon values lead to less monotonic and smooth functions. In Fig. 5, we see the effect of our perturbation on a histogram. The effect is as weak as increasing the brightness of an image by 1.

The other metrics will always have a stronger distribution shift than our metric. If we look at the first iteration step of the Insertion metric, the entire image is initially black. A

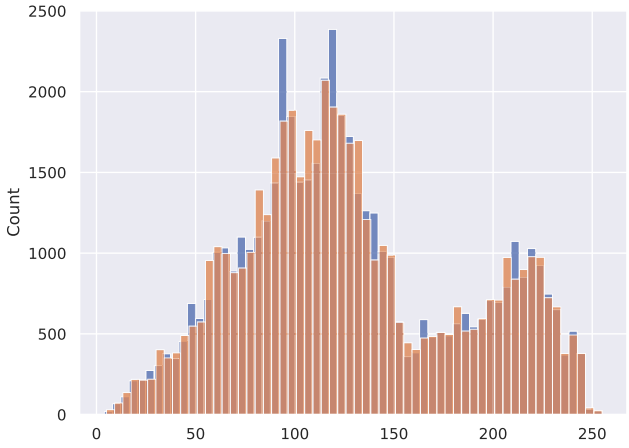


Fig. 5 Blue is the original image, orange is the perturbed image. There is only a small effect on the image distribution. Increasing the brightness by 1 would have a similar effect on the distribution (shifting it to the right). We considered here only the red channel of the violin image from Fig. 3

histogram would show a peak at 0 and nothing else, while our metric would still look like the original image.

Other attacks. We have motivated our adversarial perturbation with the FGSM attack. There are, however, 2^n other possible modifications for a given image (assuming ± 1). Many of these perturbations (such as adding -1 to all pixels) do not affect the output probability of the neural network. FGSM also does not find for every image a probability-decreasing perturbation as it relies on a single gradient ascent step. An extension to FGSM uses *projected gradient descent* (PGD) (Madry et al., 2017) to strengthen the attack, but keeps the same δ. Multiple iterations of FGSM are performed.

An important aspect of our evaluation is assessing the change in probability. Consequently, a precondition is the successful execution of the attack. Nonetheless, we have observed that weak attacks such as FGSM suffice for most datasets. It's not imperative to attain a 100% attack success rate because we can skip images that were not successfully attacked.

Baseline AMs and ground truth. For evaluation purposes, we introduce a set of synthetic baseline AMs which allows us to perform some model independent sanity checks on AM evaluation methods. The first of these baselines is called *uniform*, which emulates an equal importance of all AM pixels by sampling from the uniform distribution. Logically, evaluation methods should penalize such an input. The second baseline, which we call *canny*, applies the Canny edge detector (Canny, 1986) to the image and treats it as AM. This comparison with an edge detector was proposed in Adebayo et al. (2018). All these baseline *attribution maps* should achieve worse evaluation scores than actual AMs.

As mentioned earlier, segmentation masks are occasionally employed as a form of "ground truth" for assessing AMs. Nonetheless, a notable concern arises from the fact that not all pixels within an object hold equal significance. This phenomenon becomes evident during the prediction of a segmentation model. The central point of an object holds the highest probability, with probabilities gradually diminishing towards the object's edges. Thus, presuming a uniform probability of 100% across all segmented pixels would inaccurately reflect the decision of a neural network. This situation is even more complicated for classification models because even a single part of an object could be sufficient for determining the class.

Other scoring functions. As detailed in the section covering related work, several other scoring functions such as *Average Drop*, *Increase in confidence*, and *Coherency* rely on explanation maps. These maps involve the multiplication of the input by the saliency map. When a large portion of an image is blacked out, this induces the same distribution shift as the Deletion function. Consequently, they share the same unreliability as many other evaluation measures. In the evaluation section, we give evidence that any score function relying on masking the original image with zeros leads to unreliable results.

4 Evaluation

Given the absence of a definitive ground truth for saliency maps, the quantitative evaluation of our score function is based on consistency, mathematical properties, and comprehensive ablation studies.

To establish this, we conducted evaluations across various datasets and architectures to assess the robustness of the results. Unlike common practices in the literature, which often focus on large-scale datasets like ImageNet, we chose to incorporate real-world datasets, bolstering the credibility of our findings. Our evaluation encompassed three distinct datasets: ImageNet (Deng et al., 2009), Oxford-IIIT Pet Dataset (Parkhi et al., 2012), and ChestX-ray8 (Wang et al., 2017).

For all datasets, we analyze five CNN architectures: ResNet-50 (He et al., 2015), EfficientNet-B0 (Tan & Le, 2019), DenseNet-121 (Huang et al., 2018), ConvNeXt-Tiny (Liu et al., 2022), and RepVGG-B0 (Ding et al., 2021). The pre-trained weights were used for ImageNet, while the models were trained for the other datasets. This extensive assessment covers a total of 15 unique dataset-model combinations, constituting the most comprehensive comparison of saliency maps across diverse datasets to date.

Our evaluation implementation is based on the open *TorchCAM* (Fernandez, 2020) and *Saliency Library* (PAIR, 2022) projects.

Table 1 Evaluation scores averaged across all datasets and architectures, where Del = deletion, Ins = insertion on a black image, InsBlur = insertion on a blurred image, Perturb = FGSM perturbation

Attribution Method	Perturb (ours) ↑	InsBlur ↑	Del ↓	Ins ↑
GradCAM (Selvaraju et al., 2019)	0.477 ± 0.08	0.786 ± 0.063	0.310 ± 0.148	0.694 ± 0.109
GradCAM++ (Chattopadhay et al., 2018)	0.496 ± 0.106	0.772 ± 0.08	0.321 ± 0.168	0.679 ± 0.112
LayerCAM (Jiang et al., 2021)	0.494 ± 0.094	0.766 ± 0.055	0.337 ± 0.162	0.666 ± 0.105
PolyCAMm (Englebert et al., 2022)	0.430 ± 0.098	0.720 ± 0.064	0.315 ± 0.206	0.532 ± 0.029
PolyCAMp (Englebert et al., 2022)	0.438 ± 0.110	0.720 ± 0.073	0.318 ± 0.208	0.542 ± 0.083
PolyCAMpm (Englebert et al., 2022)	0.440 ± 0.102	0.724 ± 0.064	0.320 ± 0.203	0.546 ± 0.077
ReciproCAM (Byun & Lee, 2023)	0.495 ± 0.137	**0.794** ± 0.087	0.273 ± 0.149	**0.708** ± 0.131
ScoreCAM (Wang et al., 2020)	0.492 ± 0.113	0.764 ± 0.085	0.337 ± 0.177	0.661 ± 0.153
SmoothGradCAM++ (Omeiza et al., 2019)	0.441 ± 0.017	0.708 ± 0.040	0.427 ± 0.188	0.598 ± 0.115
BlurintegratedGradients (Xu et al., 2020)	0.454 ± 0.136	0.758 ± 0.097	**0.266** ± 0.353	0.521 ± 0.096
Gradients (Simonyan et al., 2014)	0.544 ± 0.137	0.638 ± 0.152	0.293 ± 0.314	0.371 ± 0.237
GuidedintegratedGradients (Kapishnikov et al., 2021)	0.452 ± 0.138	0.697 ± 0.116	0.294 ± 0.319	0.505 ± 0.121
IntegratedGradients (Sundararajan et al., 2017)	0.488 ± 0.135	0.712 ± 0.128	0.295 ± 0.319	0.532 ± 0.124
SmoothGrad (Smilkov et al., 2017)	**0.562** ± 0.148	0.707 ± 0.128	0.295 ± 0.319	0.561 ± 0.140
Canny (Canny, 1986)	0.315 ± 0.072	0.564 ± 0.199	0.332 ± 0.279	0.401 ± 0.211
Uniform	0.281 ± 0.061	0.528 ± 0.240	0.302 ± 0.276	0.302 ± 0.276

The first five AM methods require upsampling, while the next five methods output an AM in input resolution. The table shows that Del and Ins do not pass the baseline test, as Canny and Uniform should have the worst values. However, this table is only an overview and sanity check. We show in our evaluation section that only the ranking of Perturb is reliable

Table 2 Number of cases in which Canny edge detector and uniform came last

Method	Canny 2nd to last ↑	Uniform last ↑
Del	3	2
Ins	4	11
InsBlur	12	12
Perturb (ours)	**15**	**15**

The values marked in bold mean highest value
The maximum achievable score is 15

4.1 Quantitative Evaluation

Each dataset-model combination (e.g. DenseNet-121 with ImageNet) defines a ranking. There are in total 16 attribution maps and 15 dataset-model combinations. Since we cannot show all 15 tables, we only show the averaged results of the 15 tables in Table 1. However, we can analyze how the ranking across the different tables behaves numerically.

Baselines. Uniform and Canny serve as baseline methods designed to yield the lowest possible scores.

Across all 15 tables, we counted how often "Canny" was in second-last place and "Uniform" in last place (refer to Table 2).

Only our method passes the baseline test. InsBlur comes second but it fails for certain architectures.

Consistency. Given that attribution methods are typically conceived as general approaches, it is reasonable to anticipate a degree of similarity in their behavior across diverse datasets and architectures. Consequently, it is desirable for the metrics to yield a consistent ranking across various datasets.

To assess the degree of similarity between rankings, the rank correlation (Newson, 2002) is often used in the statistical literature. A widely used measure for this purpose is Kendall's τ. For example, if one ranking is SmoothGrad < Gradients < GradCAM++ and another ranking is SmoothGrad < GradCAM++ < Gradients, we can use this correlation measure to quantify the similarity between these two rankings.

Here we use Kendall's τ to measure the similarity across all 15 tables, resulting in the creation of a 15×15 matrix that captures the pairwise correlations between the rankings. For example, a correlation entry in this matrix would be how similar "DenseNet121 + Imagenet" and "ResNet50 + Oxford Pets" are. We compute this 15×15 matrix for all four metrics: Del, Ins, InsBlur, Perturb.

Since it would be too large to display the entire matrix, we average the entries to get an idea of the similarity. The results can be seen in Table 3. However, the complete results can be found in the appendix A.

It turns out that our metric is the most consistent of the four. Since we are comparing different architectures and datasets, it is logical that Kendall's τ is not exceptionally high.

To get a better idea of the consistency of our metric, we average the values across the architectures and look at the top 3 AMs in Table 4.

Table 3 Consistency across all model-dataset combinations

Method	Kendall's $\tau \uparrow$
Del	0.236 ± 0.409
Ins	0.353 ± 0.431
InsBlur	0.432 ± 0.314
Perturb (ours)	$\mathbf{0.466} \pm 0.252$

The values marked in bold mean highest value

Table 4 Top-3 saliency methods for the three datasets

Ranking	Oxford	ChestXray	ImageNet
1	SmoothGrad	SmoothGrad	SmoothGrad
2	Gradients	Gradients	Gradients
3	ReciproCAM	**IntGrad**	ReciproCAM

The values marked in bold mean highest value
Results are averaged across 15 tables. Only IntegratedGradients (IntGrad) differs in the ChestXray dataset

We highlighted the only attribution map that is different. Upon closer examination of all 15 dataset-architecture combinations, we found that SmoothGrad consistently claims the top position in almost all our metric's rankings. However, we would like to point out that SmoothGrad (similar to Gradients) is sometimes subject to noise. We recommend ReciproCAM or GradCAM++ as a noise-free and parameter-free alternative when noise becomes a problem. The choice of the CAM still depends on the dataset and the architecture.

Monotonicity and Smoothness. A score function exhibiting monotonic increase implies that the changes made to the pixels have not led to a distribution shift. For instance, if an image is completely obscured with black pixels and then gradually unveiled, the probability should only increase and not decrease. While revealing unimportant pixels devoid of pertinent information is possible, they should not lead to an information loss. Hence, the probability should always stay either at the same level or be higher.

In Fig. 4, we have already shown for a single image that our method leads to much smoother curves. We now extend this analysis to multiple images by defining smoothness measures.

We measure for Del the percentage of monotonic decrease and for all other evaluation functions the percentage of monotonic increase as defined by

$$\text{Monotonicity} = \frac{\sum_i \mathbf{1}[f(x_{i+1}) \geq f(x_i)]}{n-1},$$

where $1[\cdot]$ is the indicator function. Furthermore, we suggest to measure the smoothness of the functions. Various mathematical definitions of smoothness exist, most of which are not directly applicable to discrete data. In this context, we employ a definition tailored to our scenario. We compute the function's fluctuation by measuring the standard deviation of its first derivative (approximated by the forward difference). The definition is as follows:

$$\text{Smoothness} = \frac{1}{n-1}\sqrt{\sum_{i=1}^{n}(f(x_{i+1}) - f(x_i) - \overline{x})^2},$$

where $\overline{x}$ is the sample mean. The motivation for measuring smoothness is exactly the same as for monotonicity. Smoothness also implies that probability cannot fluctuate at random. As illustrated in Table 5, our metric outperforms others in both measures, highlighting its robustness.

We also want to investigate what happens if we strengthen the adversarial attack by increasing ϵ. For this purpose, we evaluate the monotonicity of our metric with respect to ϵ as seen in Fig. 6.

The results for smoothness are similar. We therefore place Fig. 12 in the appendix.

Different adversarial attacks. Adopting a stronger adversarial attack increases the number of successfully changed predictions for all images. However, if we consider the way the images are perturbed within the same gradient-based perturbation class, there is no difference. Consequently, switching to a stronger attack such as Projected Gradient

Table 5 Monotonicity and Smoothness across datasets and architectures

Method	Monotonicity $\uparrow$	Smoothness $\downarrow$
Del	0.648	1.712
Ins	0.671	1.658
InsBlur	0.654	1.238
Perturb (ours)	**0.967**	**0.891**

The values marked in bold mean highest value

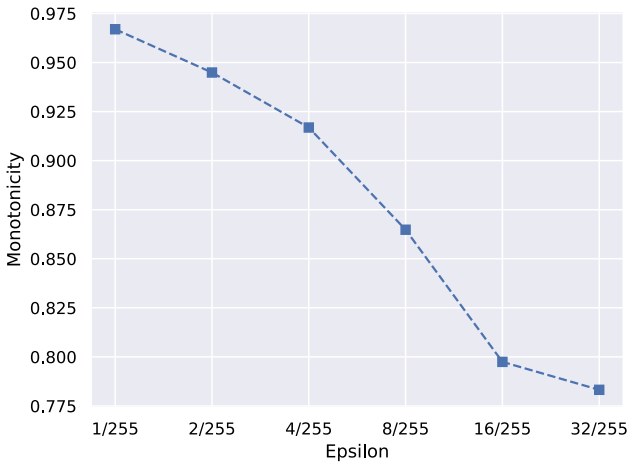


Fig. 6 This plot shows that our score function is almost always increasing at $\epsilon = \frac{1}{255}$. But with stronger ϵ, we would see fluctuations, similar to those seen with Insertion/Deletion

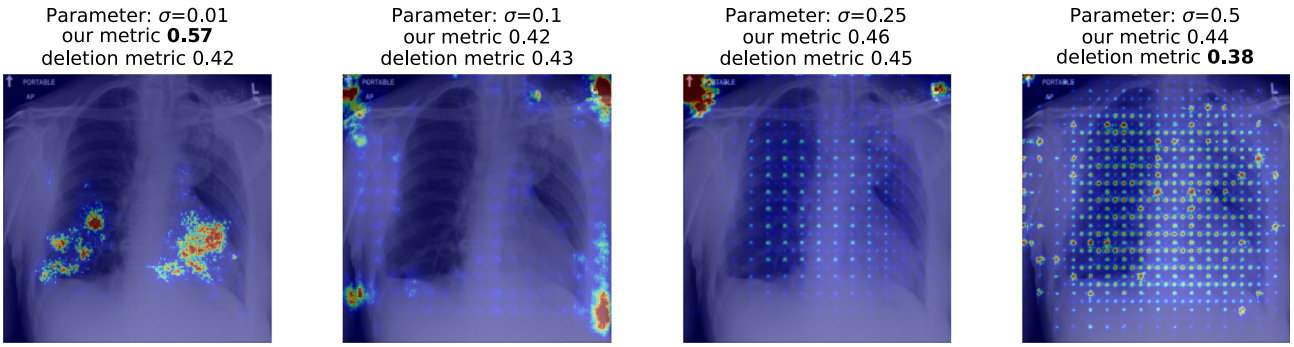


Fig. 7 The optimal σ values for SmoothGrad are determined by two evaluation methods: Deletion (↓) and our metric (↑). While visual analysis clearly suggests the first image as the most appropriate choice, the Deletion metric unexpectedly favors the last image as the preferred option

Descent (PGD) does not lead to changes in the image distribution compared to Fast Gradient Sign Method (FGSM). We systematically repeated all experiments with PGD and observed a higher success rate of the attack. For example, in the case of ChestX-ray8, the success rate increased from 95.80% to 100%. Since the perturbation remains limited to ± 1, we did not observe any change in the ranking of our scoring function. We advise resorting to PGD only when the attack success rate of FGSM proves insufficient. Other perturbation classes, such as ℓ_2, are not recommended due to their allowance of fractional pixel values, as previously discussed.

Finally, opting for a black-box attack over a gradient-based approach like FGSM comes with the drawback of potentially producing images that are more distant from the data manifold (even if the constraint is still ± 1). FGSM/PGD, being a first-order approximation, shifts the image towards a local maximum (Madry et al., 2019). Since the method makes use of the gradient of the loss function to determine the direction, it cannot move that far away from the original image.

4.2 Qualitative Evaluation

In Table 3, we observed that the evaluation function "Del" exhibited the least consistent outcomes, whereas our function displayed the most consistent results. This suggests that substantial variations in scores between different methods should be visually apparent, given the pronounced differences in their scores. For instance, if an evaluation function claims that a saliency map composed of random noise is the best, this conclusion is evidently erroneous. We have already established comparatively in Table 1 that this is indeed the case for the Deletion function (Uniform is better than Grad-CAM).

We now aim to corroborate this visual misalignment more explicitly. To achieve this, we turn our attention to a real-world application. The SmoothGrad saliency map includes a parameter for introducing Gaussian noise. The same image is processed through the neural network multiple times, each time with added noise. However, this parameter necessitates fine-tuning to attain optimal results. Traditionally, the tuning process relies on visual assessment of images while adjusting the parameter. This approach, although commonly used, lacks scientific rigor. In contrast, we propose employing our proposed metric to optimize this parameter. If an evaluation function is working as intended, then both visual and numerical assessments should yield congruent parameter values.

The outcomes are presented in Fig. 7. We evaluated four noise parameters: $\sigma \in \{0.01, 0.1, 0.25, 0.5\}$. Visually, it's evident that the first image most accurately captures the neural network's relevant features. Conversely, the third and last images display grid artifacts, indicating poor performance. Our metric selects the first image as optimal, in line with expectations. However, the Deletion metric (Del) favors the last image. Consequently, our metric also produces the anticipated qualitative outcomes.

5 Discussion and Outlook

This paper addresses the pressing challenge of evaluating attribution methods such as class activation maps (CAMs) within the context of convolutional neural networks, particularly focusing on the reliability and consistency of current evaluation measures. The observed lack of correlation in the ranking of attribution maps (AMs) across different datasets underscores the complexity of this task and the limitations of existing evaluation techniques.

Our research underscores that the unreliability of many current evaluation measures arises from the significant distribution shifts introduced by common methods like masking. These shifts disrupt the stability and accuracy of evaluation outcomes, raising the need for a more robust evaluation framework. To tackle this issue, we introduce a novel approach that replaces masking with adversarial perturba-

tions. This innovative modification maintains the original image distribution, leading to more consistent and accurate ranking of attribution methods.

Our proposed approach not only enhances the reliability of ranking but also aligns closely with expected results, providing a more trustworthy evaluation process. The demonstrated ability of our method to yield anticipated low scores for baseline methods like the Canny edge detector further supports its credibility.

Moreover, the inclusion of diverse neural network architectures in our evaluation adds to the generalizability of our findings. This observation underscores the robustness and versatility of our methodology, which has practical implications across various real-world applications.

The introduction of our perturbation-based evaluation method brings the concept of monotonicity and smoothness to the evaluation of attribution methods. The absence of significant fluctuations in the probability, as shown in our paper, further contributes to the reliability of our approach.

In terms of future outlook, there are several avenues for further research. Exploring the applicability of our method to even more complex and diverse datasets and architectures could provide deeper insights into its robustness and generalizability. For example, transformers and other sequence-based architectures could be considered for further work.

In conclusion, this paper contributes a significant step forward in addressing the challenges of evaluating attribution methods for deep learning models. By introducing a novel perturbation-based approach that addresses distribution shifts and maintains the original image distribution, we provide a more reliable and consistent framework for assessing attribution methods. This advancement holds the potential to enhance the credibility and applicability of attribution methods across diverse domains and applications in the realm of machine learning.

5.1 Limitations

A limitation of both our method and AMs in general is the case when non-existence of objects determines the CNN decision. Consider the case when the class is assigned when some object is not being in the image. Then there is no region to attack, and there are no salient pixels. Both our evaluation function and the AM would fail. Another limitation relates to the number of saliency maps used in our study. While our experiments included 16 different attribution maps, it is within the realm of possibility to extend our evaluation to more black box methods as well.

Appendix A Consistency

We computed for all four matrics (Figs. 8, 9, 10, 11) the pairwise Kendall's τ. This results in four 15×15 matrices.

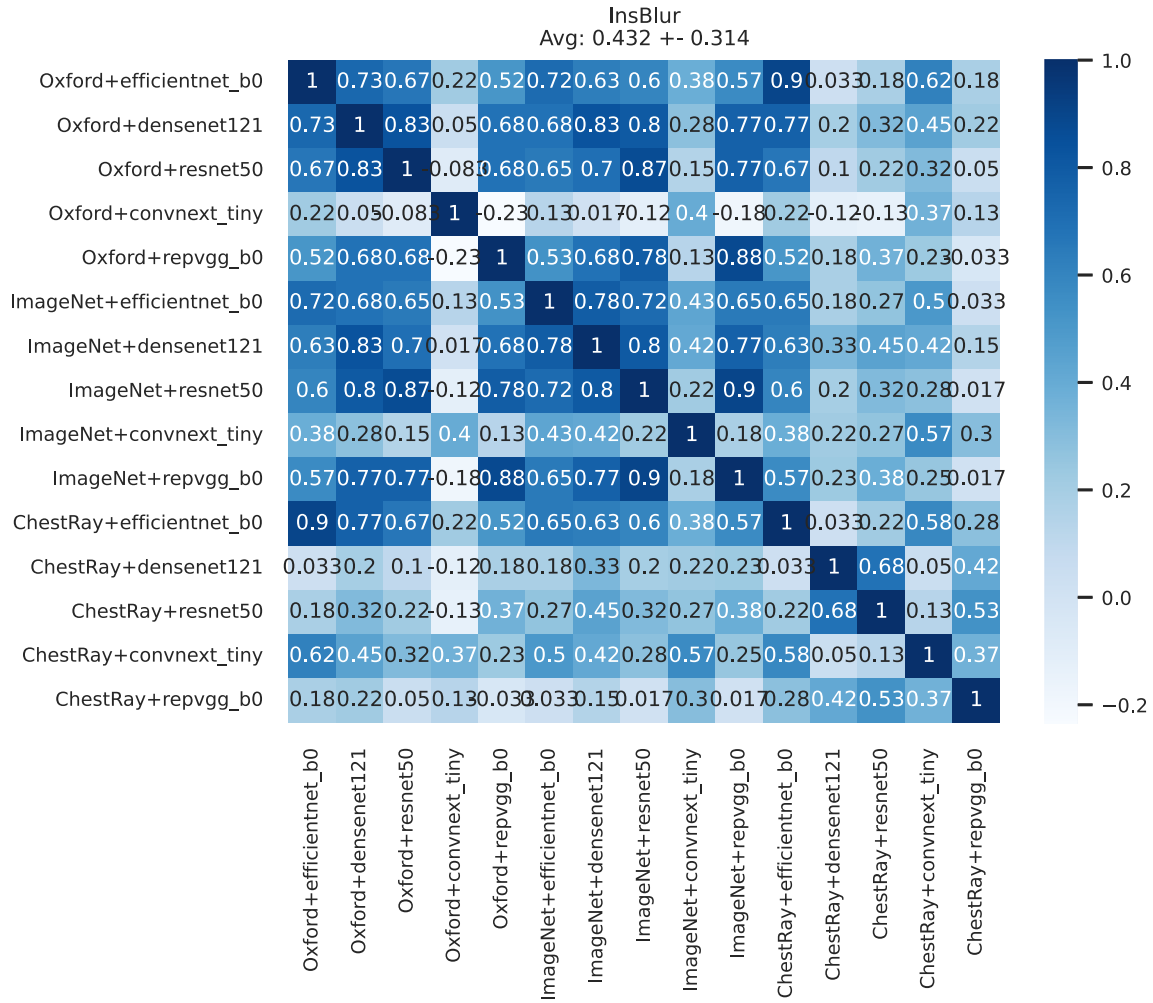


Fig. 8 Correlation matrix for InsBlur

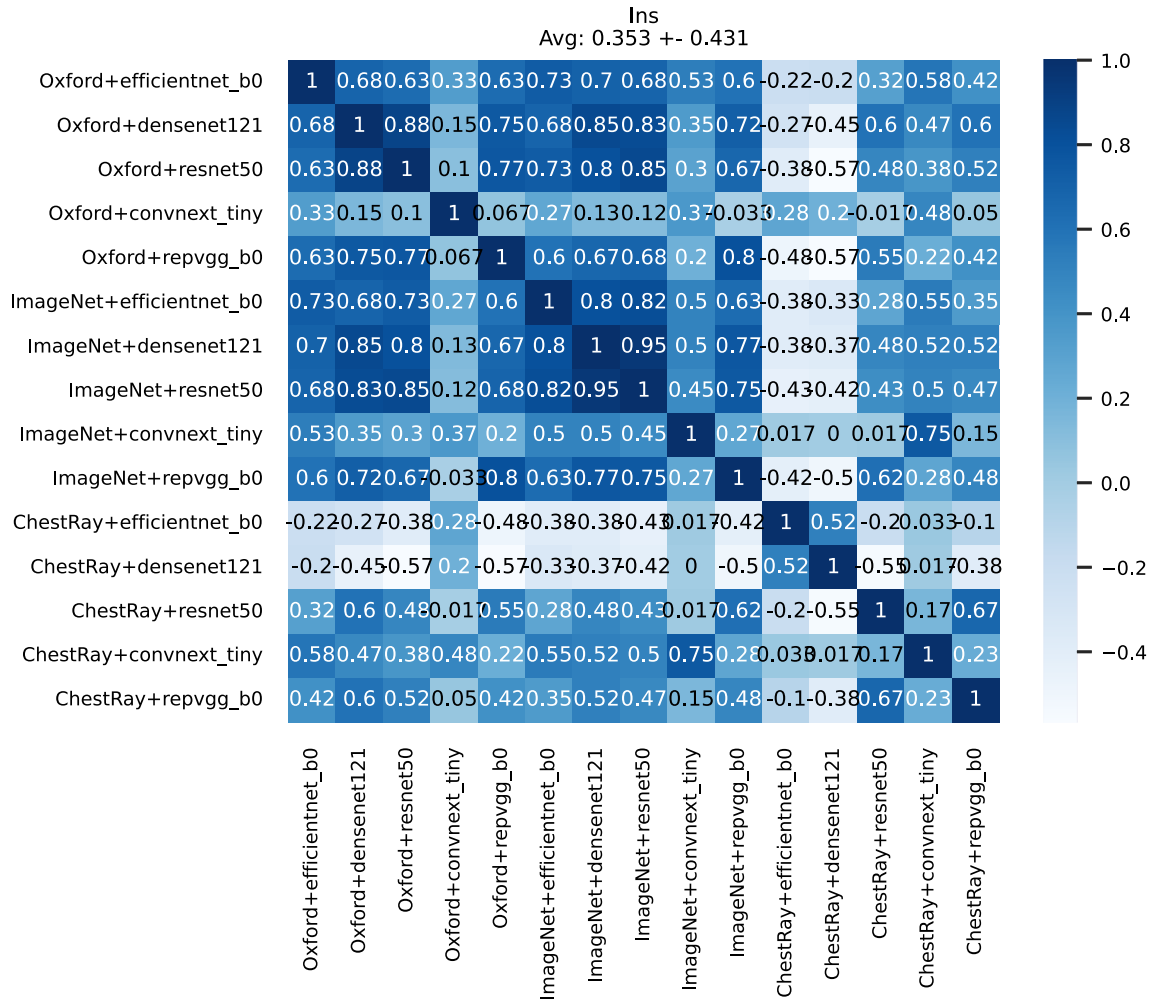


Fig. 9 Correlation matrix for Ins

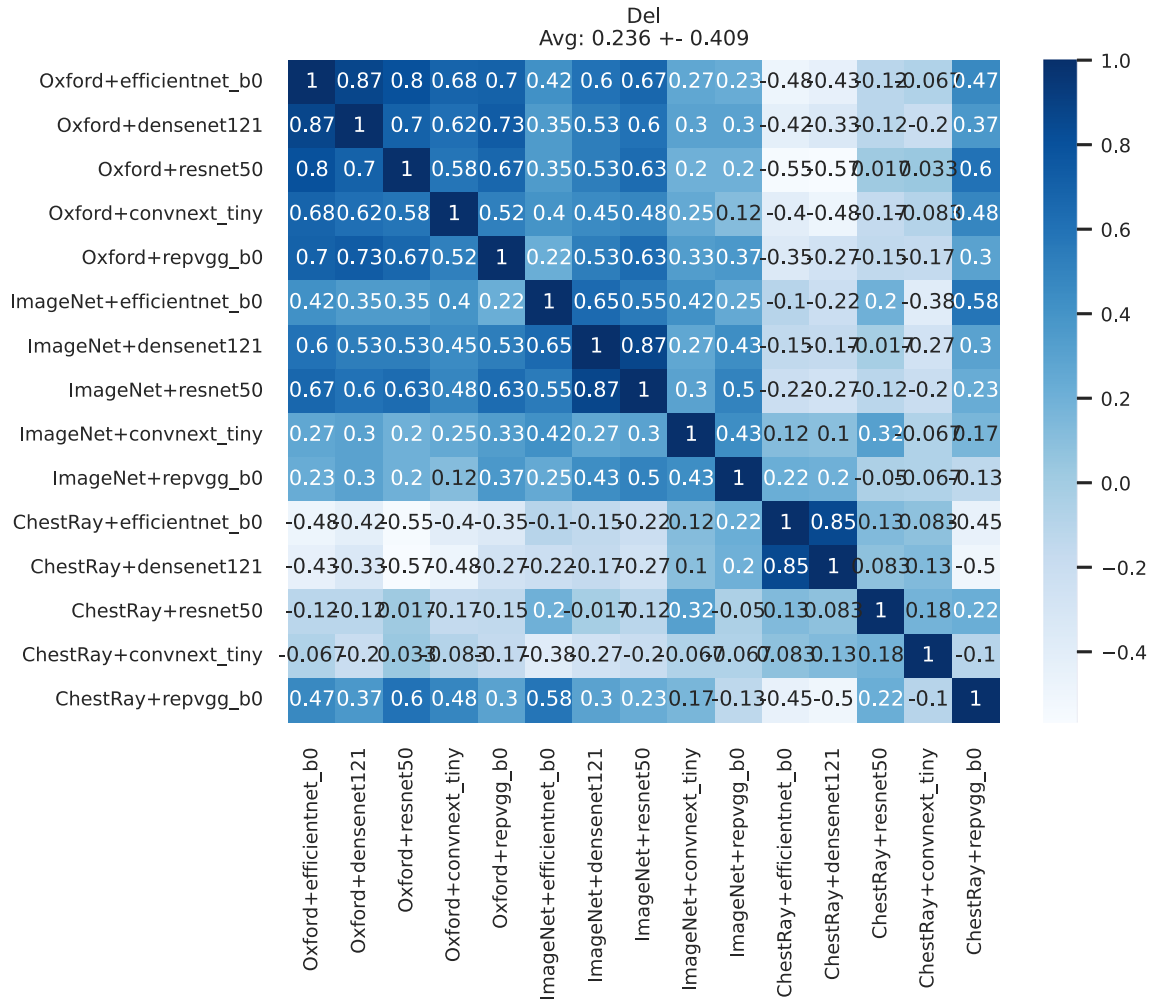


Fig. 10 Correlation matrix for Del

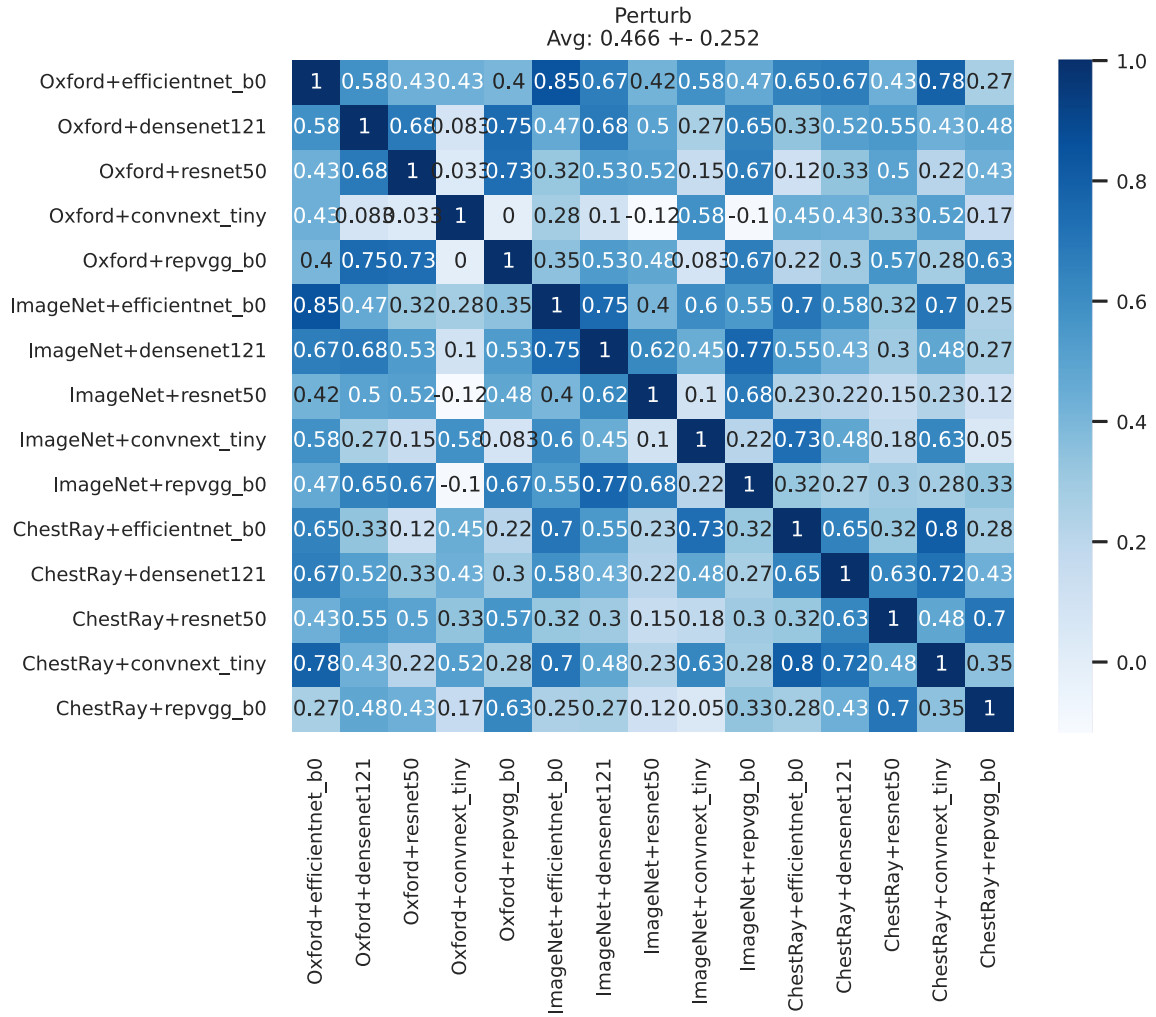


Fig. 11 Correlation matrix for Perturb (ours)

Appendix B Smoothness

The following Fig. 12 confirms the results that a lower ϵ is better.

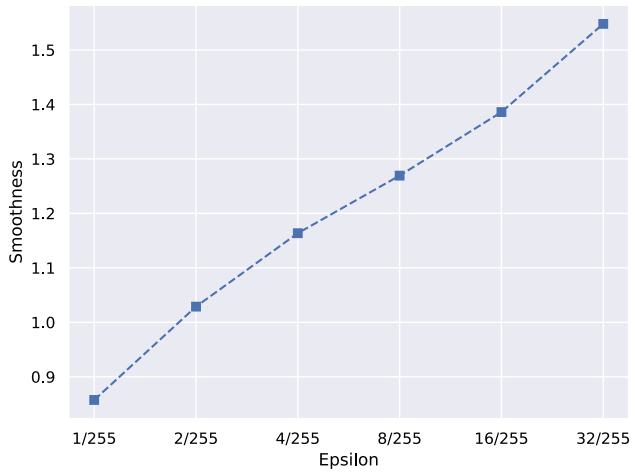


Fig. 12 This plot shows the effect of ϵ on smoothness. Lower is better

Funding Open Access funding enabled and organized by Projekt DEAL.

Data Availability All the datasets used in this paper are available online: The Oxford-IIIT Pet Dataset (https://www.robots.ox.ac.uk/\protect\unhbox\voidb@x\penalty\@M\vgg/data/pets/), ImageNet (https://www.image-net.org/), ChestX-ray8 (https://redivis.com/datasets/612s-dx4rxexky).

Declarations

Conflict of interest The author(s) declare that they have no Conflict of interest to disclose.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the

permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., & Kim, B. (2018). Sanity checks for saliency maps. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in neural information processing systems, vol. 31. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2018/file/294a8ed24b1ad22ec2e7efea049b8737-Paper.pdf

Akhtar, N., & Mian, A. (2018). Threat of adversarial attacks on deep learning in computer vision: A survey. *IEEE Access, 6*, 14410–14430.

Ancona, M., Ceolini, E., Öztireli, A.C., & Gross, M.H. (2017). A unified view of gradient-based attribution methods for deep neural networks. arXiv:1711.06104

Byun, S.-Y., & Lee, W. (2023). Recipro-CAM: Fast gradient-free visual explanations for convolutional neural networks

Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 6*, 679–698.

Carlini, N., & Wagner, D.A. (2016). Towards evaluating the robustness of neural networks. arXiv:1608.04644

Chattopadhay, A., Sarkar, A., Howlader, P., & Balasubramanian, V.N. (2018). Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks. In: 2018 IEEE winter conference on applications of computer vision (WACV). IEEE. https://doi.org/10.1109/wacv.2018.00097

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition, pp. 248–255 . https://doi.org/10.1109/CVPR.2009.5206848

Desai, S., & Ramaswamy, H.G. (2020). Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. In: 2020 IEEE winter conference on applications of computer vision (WACV), pp. 972–980. https://doi.org/10.1109/WACV45572.2020.9093360

Ding, X., Zhang, X., Ma, N., Han, J., Ding, G., & Sun, J. (2021). Repvgg: Making VGG-style convnets great again. arXiv:2101.03697

Englebert, A., Cornu, O., & De Vleeschouwer, C. (2022). Poly-CAM: High resolution class activation map for convolutional neural networks. arXiv. https://doi.org/10.48550/ARXIV.2204.13359 . arXiv:2204.13359

Everingham, M., Gool, L. V., Williams, C. K. I., Winn, J., & Zisserman, A. (2009). The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision, 88*(2), 303–338. https://doi.org/10.1007/s11263-009-0275-4

Fernandez, F.-G. (2020). TorchCAM: Class activation explorer. GitHub.

Fong, R., & Vedaldi, A. (2017). Interpretable explanations of black boxes by meaningful perturbation. arXiv:1704.03296

Fong, R., Patrick, M., & Vedaldi, A. (2019). Understanding deep networks via extremal perturbations and smooth masks. CoRR arXiv:1910.08485

Fu, R., Hu, Q., Dong, X., Guo, Y., Gao, Y., & Li, B. (2020). Axiom-based grad-cam: Towards accurate visualization and explanation of cnns. CoRR arXiv:2008.02312

Gilpin, L.H., Bau, D., Yuan, B.Z., Bajwa, A., Specter, M., & Kagal, L. (2018). Explaining explanations: An overview of interpretability of machine learning. In: 2018 IEEE 5th international conference on data science and advanced analytics (DSAA), pp. 80–89. IEEE

Gomez, T., Fréour, T., & Mouchère, H. (2022). Metrics for saliency map evaluation of deep learning explanation methods. CoRR arXiv:2201.13291

Goodfellow, I.J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. arXiv . https://doi.org/10.48550/ARXIV.1412.6572 . https://arxiv.org/abs/1412.6572

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. CoRR arXiv:1512.03385

Huang, G., Liu, Z., Weinberger, K.Q. (2018). Densely connected convolutional networks. CoRR arXiv:1608.06993

Jiang, P.-T., Zhang, C.-B., Hou, Q., Cheng, M.-M., & Wei, Y. (2021). Layercam: Exploring hierarchical class activation maps for localization. *IEEE Transactions on Image Processing, 30*, 5875–5888. https://doi.org/10.1109/TIP.2021.3089943

Kapishnikov, A., Venugopalan, S., Avci, B., Wedin, B., Terry, M., & Bolukbasi, T. (2021). Guided integrated gradients: An adaptive path method for removing noise. CoRR arXiv:2106.09788

LeCun, Y., Bengio, Y., et al. (1998). Convolutional networks for images, speech, and time series. In: The handbook of brain theory and neural networks. Convolutional networks for images, speech, and time series, pp. 255–258. MIT Press, Cambridge, MA, USA.

Li, Y., Cheng, M., Hsieh, C., & Lee, T.C.M. (2021). A review of adversarial attack and defense for classification methods. CoRR arXiv:2111.09961

Li, H., Li, Z., Ma, R., & Wu, T. (2022). FD-CAM: Improving faithfulness and discriminability of visual explanation for CNNs. arXiv. https://doi.org/10.48550/ARXIV.2206.08792 . arXiv:2206.08792

Liang, B., Li, H., Su, M., Li, X., Shi, W., & Wang, X. (2021). Detecting adversarial image examples in deep neural networks with adaptive noise reduction. *IEEE Transactions on Dependable and Secure Computing, 18*(1), 72–85. https://doi.org/10.1109/TDSC.2018.2874243

Lin, T., Maire, M., Belongie, S.J., Bourdev, L.D., Girshick, R.B., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C.L. (2014). Microsoft COCO: Common objects in context. CoRR arXiv:1405.0312

Liu, Z., Mao, H., Wu, C., Feichtenhofer, C., Darrell, T., & Xie, S. (2022). A convnet for the 2020s. CoRR arXiv:2201.03545

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. arXiv. https://doi.org/10.48550/ARXIV.1706.06083 . arXiv:1706.06083

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2019). Towards deep learning models resistant to adversarial attacks.

Muhammad, M.B., & Yeasin, M. (2020). Eigen-cam: Class activation map using principal components. CoRR arXiv:2008.00299

Naidu, R., Ghosh, A., Maurya, Y., K, S.R.N., & Kundu, S.S. (2020). IS-CAM: Integrated score-cam for axiomatic-based explanations. CoRR arXiv:2010.03023

Newson, R. (2002). Parameters behind "nonparametric" statistics: Kendall's tau, somers' d and median differences. *The Stata Journal: Promoting communications on statistics and Stata, 2*(1), 45–64. https://doi.org/10.1177/1536867x0200200103

Nieradzik, L., Stephani, H., Sieburg-Rockel, J., Helmling, S., Olbrich, A., & Keuper, J. (2024). Challenging the black box: A comprehensive evaluation of attribution maps of CNN applications in agriculture and forestry. In: *Proceedings of the 19th international joint conference on computer vision, imaging and computer graphics theory and applications*, Volume 2: VISAPP, 2024, pp. 483–492

Omeiza, D., Speakman, S., Cintas, C., & Weldemariam, K. (2019). Smooth grad-cam++: An enhanced inference level visualization technique for deep convolutional neural network models. CoRR arXiv:1908.01224

PAIR, G. (2022). Saliency library. GitHub

Parkhi, O.M., Vedaldi, A., Zisserman, A., & Jawahar, C.V. (2012). Cats and dogs. In: IEEE conference on computer vision and pattern recognition.

Petsiuk, V., Das, A., & Saenko, K. (2018). RISE: Randomized input sampling for explanation of black-box models. arXiv . https://doi.org/10.48550/ARXIV.1806.07421 . arXiv:1806.07421

Petsiuk, V., Das, A., & Saenko, K. (2018). RISE: Randomized input sampling for explanation of black-box models. CoRR arXiv:1806.07421

Poppi, S., Cornia, M., Baraldi, L., & Cucchiara, R. (2021). Revisiting the evaluation of class activation mapping for explainability: A novel metric and experimental analysis. CoRR arXiv:2104.10252

Raatikainen, L., & Rahtu, E. (2022). The weighting game: Evaluating quality of explainability methods. arXiv . https://doi.org/10.48550/ARXIV.2208.06175 . arXiv:2208.06175

Rauber, J., Brendel, W., & Bethge, M. (2017). Foolbox v0.8.0: A python toolbox to benchmark the robustness of machine learning models. CoRR arXiv:1707.04131

Ribeiro, M.T., Singh, S., & Guestrin, C. (2016). "why should I trust you?": Explaining the predictions of any classifier. CoRR arXiv:1602.04938

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2019). Grad-CAM: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision, 128*(2), 336–359. https://doi.org/10.1007/s11263-019-01228-7

Shi, X., Khademi, S., Li, Y., & Gemert, J. (2020). Zoom-cam: Generating fine-grained pixel annotations from image labels. CoRR arXiv:2010.08644

Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Deep inside convolutional networks: Visualising image classification models and saliency maps. In: Bengio, Y., LeCun, Y. (eds.) 2nd international conference on learning representations, ICLR 2014, Banff, AB, Canada, Apr 14–16, 2014, Workshop track proceedings. arXiv:1312.6034

Smilkov, D., Thorat, N., Kim, B., Viégas, F.B., & Wattenberg, M. (2017). Smoothgrad: Removing noise by adding noise. CoRR arXiv:1706.03825

Springenberg, J.T., Dosovitskiy, A., Brox, T., & Riedmiller, M.A. (2015). Striving for simplicity: The all convolutional net. In: Bengio, Y., LeCun, Y. (eds.) 3rd International conference on learning representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Workshop track proceedings. arXiv:1412.6806

Su, J., Vargas, D.V., & Sakurai, K. (2017). One pixel attack for fooling deep neural networks. CoRR arXiv:1710.08864

Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. CoRR arXiv:1703.01365 .

Tan, M., & Le, Q.V. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. CoRR arXiv:1905.11946

Wang, H., et al. (2020) Score-CAM: Score-weighted visual explanations for convolutional neural networks. In: *2020 IEEE/CVF conference on computer vision and pattern recognition workshops (CVPRW)*, pp. 111–119. Seattle, WA, USA. https://doi.org/10.1109/CVPRW50498.2020.00020.

Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., & Summers, R.M. (2017). Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In: CVPR, pp. 3462–3471. IEEE Computer Society. http://dblp.uni-trier.de/db/conf/cvpr/cvpr2017.html#WangPLLBS17

Wang, H., Wang, Z., Du, M., Yang, F., Zhang, Z., Ding, S., Mardziel, P., & Hu, X. (2020). Score-CAM: Score-weighted visual explanations for convolutional neural networks.

Xu, W., Evans, D., & Qi, Y. (2017). Feature squeezing: Detecting adversarial examples in deep neural networks. CoRR arXiv:1704.01155

Xu, S., Venugopalan, S., & Sundararajan, M. (2020). Attribution in scale and space. CoRR arXiv:2004.03383

Zarándy, Á., Rekeczky, C., Szolgay, P., & Chua, L.O. (2015). Overview of cnn research: 25 years history and the current trends. In: 2015 IEEE international symposium on circuits and systems (ISCAS), pp. 401–404. IEEE

Zeiler, M.D., & Fergus, R. (2013). Visualizing and understanding convolutional networks. CoRR arXiv:1311.2901

Zhang, J., Lin, Z., Brandt, J., Shen, X., & Sclaroff, S. (2016). Top-down neural attention by excitation backprop. CoRR arXiv:1608.00507

Zhang, Q., Rao, L., & Yang, Y. (2021). Group-cam: Group score-weighted visual explanations for deep convolutional networks. CoRR arXiv:2103.13859

Zhang, Y., Tiňo, P., Leonardis, A., & Tang, K. (2021). A survey on neural network interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence, 5*(5), 726–742.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.