This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Towards Better Understanding Attribution Methods

Sukrut Rao, Moritz Böhle, Bernt Schiele
Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany
{sukrut.rao,mboehle,schiele}@mpi-inf.mpg.de

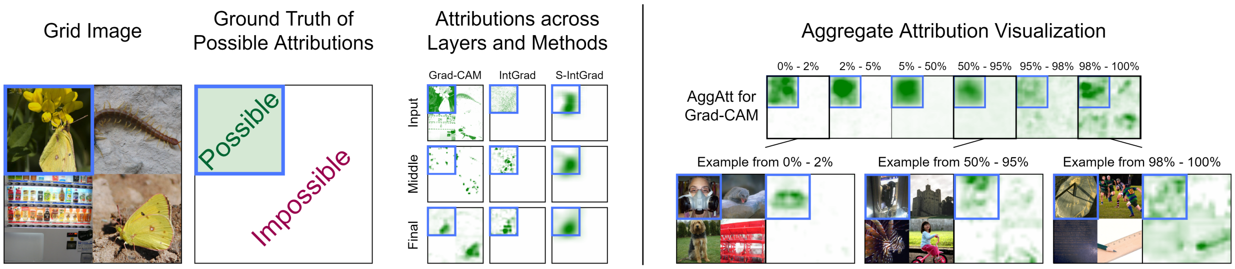


Fig. 1. **Left:** Illustration of *DiFull* and *ML-Att*. In DiFull, we evaluate models on image grids (col. 1). Crucially, we employ separate *classification heads* for each subimage that cannot possibly be influenced by other subimages; this yields 'ground truths' for possible and impossible attributions (col. 2). For ML-Att, we evaluate methods at different network layers; here we show results for Grad-CAM and IntGrad. Further, we show results after smoothing IntGrad (S-IntGrad), which we find to perform well (Sec. 5.2). **Right:** Visualisation of our AggAtt evaluation. By sorting attributions into percentile ranges w.r.t. their performance and aggregating them over many samples, we obtain a holistic view of a methods' performance. AggAtt can thus reflect both best and worst case behaviour of an attribution method.

Abstract

Deep neural networks are very successful on many vision tasks, but hard to interpret due to their black box nature. To overcome this, various post-hoc attribution methods have been proposed to identify image regions most influential to the models' decisions. Evaluating such methods is challenging since no ground truth attributions exist. We thus propose three novel evaluation schemes to more reliably measure the faithfulness *of those methods, to make comparisons between them more* fair*, and to make* visual inspection *more* systematic. *To address faithfulness, we propose a novel evaluation setting (DiFull) in which we carefully control which parts of the input can influence the output in order to distinguish possible from impossible attributions. To address fairness, we note that different methods are applied at different layers, which skews any comparison, and so evaluate all methods on the same layers (ML-Att) and discuss how this impacts their performance on quantitative metrics. For more systematic visualizations, we propose a scheme (AggAtt) to qualitatively evaluate the methods on complete datasets. We use these evaluation schemes to study strengths and shortcomings of some widely used attribution methods. Finally, we propose a post-processing smoothing step that significantly improves the performance of some attribution methods, and discuss its applicability.*

1. Introduction

Deep neural networks (DNNs) are highly successful on many computer vision tasks. However, their black box nature makes it hard to interpret and thus trust their decisions. To shed light on the models' decision-making process, several methods have been proposed that aim to attribute importance values to individual input features (see Sec. 2). However, given the lack of ground truth importance values, it has proven difficult to compare and evaluate these *attribution methods* in a holistic and systematic manner.

In this work, we take a three-pronged approach towards addressing this issue. In particular, we focus on three important components for such evaluations: reliably measuring the methods' *model-faithfulness*, ensuring a *fair comparison* between methods, and providing a framework that allows for *systematic* visual inspections of their attributions.

First, we propose an evaluation scheme (**DiFull**), which allows distinguishing possible from impossible importance attributions. This effectively provides ground truth annotations for whether or not an input feature can possibly have influenced the model output. As such, it can highlight distinct failure modes of attribution methods (Fig. 1, left).

Second, a fair evaluation requires attribution methods to be compared on equal footing. However, we observe that different methods explain DNNs to different depths (e.g., full network or classification head only). Thus, some methods in fact solve a much easier problem (i.e., explain a much shallower network). To even the playing field, we propose a multi-layer evaluation scheme for attributions (**ML-Att**) and provide a thorough evaluation of commonly used methods across multiple layers and models (Fig. 1, left). When compared on the same level, we find that performance differences between some methods essentially vanish.

Third, relying on individual examples for a qualitative comparison is prone to skew the comparison and cannot fully represent the evaluated attribution methods. To overcome this, we propose a qualitative evaluation scheme for which we aggregate attribution maps (**AggAtt**) across many input samples. This allows us to observe trends in the performance of attribution methods across complete datasets, in addition to looking at individual examples (Fig. 1, right).

Contributions. (1) We propose a novel evaluation setting, **DiFull**, in which we control which regions *cannot possibly* influence a model's output, which allows us to highlight definite failure modes of attribution methods. **(2)** We argue that methods can only be compared fairly when evaluated *on the same layer*. To do this, we introduce **ML-Att** and evaluate all attribution methods at multiple layers. We show that, when compared fairly, apparent performance differences between some methods effectively vanish. **(3)** We propose a novel aggregation method, **AggAtt**, to qualitatively evaluate attribution methods across all images in a dataset. This allows to qualitatively assess a method's performance across many samples (Fig. 1, right), which complements the evaluation on individual samples. **(4)** We propose a post-processing smoothing step that significantly improves localization performance on some attribution methods. We observe significant differences when evaluating these smoothed attributions on different architectures, which highlights how architectural design choices can influence an attribution method's applicability. Our code is available at https://github.com/sukrutrao/Attribution-Evaluation.

2. Related Work

Post-hoc attribution methods broadly use one of three main mechanisms. *Backpropagation-based* methods [26, 27, 30–32, 35] typically rely on the gradients with respect to the input [26, 27, 30, 32] or with respect to intermediate layers [17,35]. *Activation-based* methods [6,8,12,23,33,36] weigh activation maps to assign importance, typically of the final convolutional layer. The activations may be weighted by their gradients [6,12,23,36] or by estimating their importance to the classification score [8, 33]. *Perturbation-based* methods [7, 9, 16, 18, 34] treat the network as a black-box and assign importance by observing the change in output on perturbing the input. This is done by occluding parts of the image [16, 18, 34] or optimizing for a mask that maximizes/minimizes class confidence [7, 9].

In this work, we evaluate on a diverse set of attribution methods spanning all three categories.

Evaluation Metrics: Several metrics have been proposed to evaluate attribution methods, and can broadly be categorised into *Sanity checks*, *localization-*, and *perturbation-based metrics*. Sanity checks [1, 2, 17] test for basic properties an attribution method must satisfy (e.g., the explanation should depend on the model parameters). Localization-based metrics evaluate how well attributions localize class discriminative features of the input. Typically, this is done by measuring how well attributions coincide with object bounding boxes or image grid cells (see below) [4, 5, 9, 22, 35]. Perturbation-based metrics measure model behaviour under input perturbation to estimate feature importance. Examples include removing the most [21] or least [31] salient pixels, or using the attributions to scale input features and measuring changes in confidence [6]. Our work combines aspects from localization metrics and sanity checks to evaluate the model-faithfulness of an attribution method.

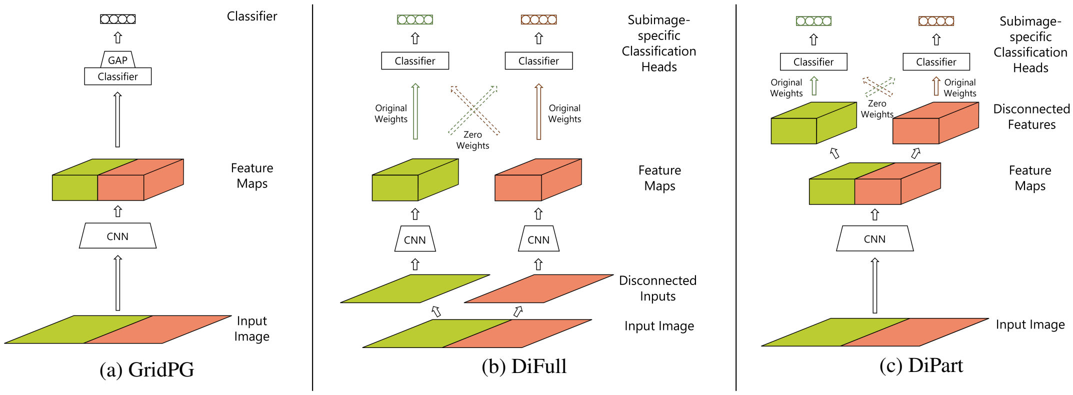
Localization on Grids: Relying on object bounding boxes for localization assumes that the model only relies on information *within* those bounding boxes. However, neural networks are known to also rely on context information for their decisions, cf. [25]. Therefore, recent work [3, 4, 24] proposes creating a grid of inputs from distinct classes and measuring localization to the entire grid cell, which allows evaluation on datasets where bounding boxes are not available. However, this does not *guarantee* that the model only uses information from within the grid cell, and may fail for similar looking features (Fig. 3, right). In our work, we propose a metric that controls the flow of information and guarantees that grid cells are classified independently.

3. Evaluating Attribution Methods

We present our evaluation settings for better understanding the strengths and shortcomings of attribution methods. Similar to the Grid Pointing Game (GridPG) [4], these metrics evaluate attribution methods on image grids with multiple classes. In particular, we propose a novel quantitative metric, DiFull, and extension to it, DiPart (3.1), as stricter tests of model faithfulness than GridPG. Further, we present a qualitative metric, AggAtt (3.2) and an evaluation setting that compares methods at identical layers, ML-Att (3.3)

3.1. Quantitative Evaluation: Disconnecting Inputs

In the following, we introduce the quantitative metrics that we use to compare attribution methods. For this, we first describe GridPG and the grid dataset construction it uses [4]. We then devise a novel setting, in which we carefully control which features can influence the model output. By construction, this provides ground truth annotations for image regions that can or cannot possibly have influenced the model output. While GridPG evaluates how well the methods localize class discriminative features, our metrics



(a) GridPG (b) DiFull (c) DiPart

Fig. 2. **Our three evaluation settings.** In GridPG, the classification scores are influenced by the entire input. In DiFull, on the other hand, we explicitly control which inputs can influence the classification score. For this, we pass each subimage separately through the spatial layers, and then construct individual classification heads for each of the subimages. DiPart serves as a more natural setting to DiFull, that still provides partial control over information. We show a 1×2 grid for readability, but the experiments use 2×2 grids.

complement it by evaluating their *model-faithfulness*.

3.1.1 Grid Data and GridPG

For GridPG [4], the attribution methods are evaluated on a synthetic grid of $n \times n$ images in which each class may occur at most once. In particular, for each of the occurring classes, GridPG measures the fraction of positive attribution assigned to the respective grid cell versus the overall amount of positive attribution. Specifically, let $A^+(p)$ refer to the positive attribution given to the p^{th} pixel. The localization score for the subimage x_i is given by:

$$L_i = \frac{\sum_{p \in x_i} A^+(p)}{\sum_{j=1}^{n^2} \sum_{p \in x_j} A^+(p)} \qquad (1)$$

An 'optimal' attribution map would thus yield $L_i\!=\!1$, while uniformly distributing attributions would yield $L_i\!=\!\frac{1}{n^2}$.

By only using confidently classified images from distinct classes, GridPG aims to ensure that the model does not find 'positive evidence' for any of the occurring classes in the grid cells of other classes. However, specifically for class-combinations that share low-level features, this assumption might not hold, see Fig. 3 (right): despite the two dogs (upper left and lower right) being classified correctly as single images, the output for the logit of the dog in the upper left is influenced by the features of the dog in the lower right in the grid image. Since all images in the grid can indeed influence the model output in GridPG[1], it is unclear whether such an attribution is in fact not *model-faithful*.

3.1.2 Proposed Metric: DiFull

As discussed, the assumption in GridPG that no feature outside the subimage of a given class should positively influence the respective class logit might not hold. Hence, we propose to fully disconnect (DiFull) the individual subimages from the model outputs for other classes. For this, we introduce two modifications. First, after removing the GAP operation, we use $n \times n$ classification heads, one for each subimage, only *locally* pooling those outputs that have their receptive field center *above the same subimage*. Second, we ensure that their receptive field does not overlap with other subimages by zeroing out the respective connections.

In particular, we implement DiFull by passing the subimages separately through the CNN backbone of the model under consideration[2], see Fig. 2b. Then, we apply the classification head separately to the feature maps of each subimage. As we discuss in the supplement, DiFull has similar computational requirements as GridPG.

As a result, we can *guarantee* that no feature outside the subimage of a given class can possibly have influenced the respective class logit—they are indeed *fully disconnected*.

3.1.3 Natural Extension: DiPart

At one end, GridPG allows any subimage to influence the output for any other class, while at the other, DiFull completely disconnects the subimages. In contrast to GridPG, DiFull might be seen as a constructed setting not seen in typical networks. As a more natural setting, we therefore propose DiPart, for which we only partially disconnect the subimages from the outputs for other classes, see Fig. 2c. Specifically, we do not zero out all connections (Sec. 3.1.2), but instead only apply the local pooling operation from DiFull and thus obtain local classification heads for each subimage (as in DiFull). However, in this setting, the classification head for a specific subimage can be influenced by features in other subimages that lie within the head's receptive field. For models with a small receptive field, this yields

[1] As shown in Fig. 2a, the convolutional layers of the model under consideration process the entire grid to obtain feature maps, which are then classified point-wise. Finally, a *single output per class* is obtained by globally pooling all point-wise classification scores. As such, the class logits can, of course, be influenced by all images in the grid.

[2] Note that this is equivalent to setting the respective weights of a convolutional kernel to zero every time it overlaps with another subimage.

very similar results as DiFull (Sec. 5 and Supplement).

3.2. Qualitative Evaluation: AggAtt

In addition to quantitative metrics, attribution methods are often compared qualitatively on individual examples for a visual assessment. However, this is sensitive to the choice of examples and does not provide a holistic view of the method's performance. By constructing standardised grids, in which 'good' and 'bad' (GridPG) or *possible* and *impossible* (DiFull) attributions are always located in the same regions, we can instead construct *aggregate attribution maps*.

Thus, we propose a new qualitative evaluation scheme, **AggAtt**, for which we generate a set of aggregate maps for each method that progressively show the performance of the methods from the best to the worst localized attributions.

For this, we first select a grid location and then sort all corresponding attribution maps in descending order of the localization score, see Eq. (1). Then, we bin the maps into percentile ranges and, finally, obtain an aggregate map per bin by averaging all maps within a single bin. In our experiments, we observed that attribution methods typically performed consistently over a wide range of inputs, but showed significant deviations in the tails of the distributions (best and worst case examples). Therefore, to obtain a succinct visualization that highlights both distinct failure cases as well as the best possible results, we use bins of unequal sizes. Specifically, we use smaller bins for the top and bottom percentiles. For an example of AggAtt, see Fig. 1.

As a result, AggAtt allows for a *systematic* qualitative evaluation and provides a holistic view of the performance of attribution methods across many samples.

3.3. Attributions Across Network Layers: ML-Att

Attribution methods often vary significantly in the degree to which they explain a model. Activation-based attribution methods like Grad-CAM [23], e.g., are typically applied on the last spatial layer, and thus only explain a fraction of the full network. This is a significantly easier task as compared to explaining the entire network, as is done by typical backpropagation-based methods. Activations from deeper layers of the network would also be expected to localize better, since they would represent the detection of higher level features by the network (Fig. 1, left).

For a *fair comparison* between methods, we thus propose a multi-layer evaluation scheme for attributions (**ML-Att**). Specifically, we evaluate methods at various network layers and compare their performance *on the same layers*. For this, we evaluate all methods at the input, an intermediate, and the final spatial layer of multiple network architectures, see Sec. 4 for details. Importantly, we find that apparent differences found between some attribution methods vanish when compared *fairly*, i.e., on the same layer (Sec. 5.1).

Lastly, we note that most attribution methods have been designed to assign importance values to *input features* of the model, not *intermediate* network activations. The generalisation to intermediate layers, however, is straightforward. For this, we simply divide the full model $\mathbf{f}_{\text{full}}$ into two virtual parts: $\mathbf{f}_{\text{full}} = \mathbf{f}_{\text{explain}} \circ \mathbf{f}_{\text{pre}}$. Specifically, we treat $\mathbf{f}_{\text{pre}}$ as a pre-processing step and use the attribution methods to explain the outputs of $\mathbf{f}_{\text{explain}}$ with respect to the inputs $\mathbf{f}_{\text{pre}}(\mathbf{x})$. Note that in its standard use case, in Grad-CAM $\mathbf{f}_{\text{pre}}(\mathbf{x})$ is given by all convolutional layers of the model, whereas for most gradient-based methods $\mathbf{f}_{\text{pre}}(\mathbf{x})$ is the identity.

4. Experimental Setup

Dataset and Architectures: We run our experiments on VGG11 [28] and Resnet18 [10] trained on Imagenet [19]; similar results were observed on CIFAR10 [14] (in supplement). For each model, we separately select images from the validation set that were classified with a confidence score of at least 0.99. By only using highly confidently classified images [3, 4], we ensure that the features within each grid cell constitute positive evidence of its class for the model, and features outside it contain low positive evidence since they get confidently classified to a different class.

Evaluation on GridPG, DiFull, and DiPart: We evaluate on 2×2 grids constructed by randomly sampling images from the set of confidently classified images (see above). Specifically, we generate 2000 attributions per method for each of GridPG, DiFull, and DiPart. For GridPG, we use images from distinct classes, while for DiFull and DiPart we use distinct classes except in the bottom right corner, where we use the same class as the top left. By repeating the same class twice, we can test whether an attribution method simply highlights class-related features, irrespective of them being used by the model. Since subimages are disconnected from the classification heads of other locations in DiFull and DiPart, the use of repeating classes does not change which regions should be attributed (Sec. 3.1.2).

Evaluation at Intermediate Layers: We evaluate each method at the input (image), middle[3] (Conv5 for VGG11, Conv3_x for Resnet18), and final spatial layer (Conv8 for VGG11, Conv5_x for Resnet18) of each network, see Sec. 3.3. Evaluating beyond the input layer leads to lower dimensional attribution maps, given by the dimensions of the activation maps at those layers. Thus, as is common practice [23], we upsample those maps to the dimensions of the image (448 × 448) using bilinear interpolation.

Qualitative Evaluation on AggAtt: As discussed, for AggAtt we use bins of unequal sizes (Sec. 3.2). In particular, we bin the attribution maps into the following percentile ranges: 0–2%, 2–5%, 5–50%, 50–95%, 95–98%, and 98–100%; cf. Fig. 1. Further, in our experiments we evaluate the attributions for classes at the top-left grid location.

[3] We show a single intermediate layer to visualize trends from the input to the final layer. Results on all layers can be found in the supplement.

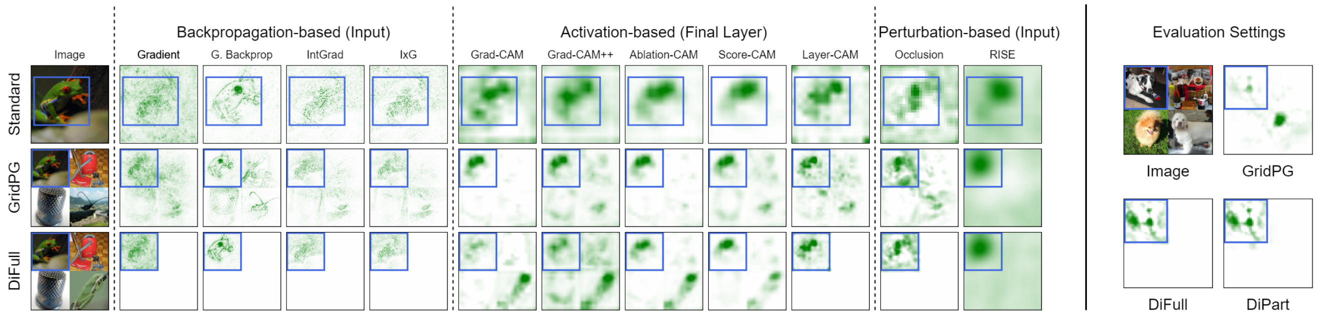


Fig. 3. **Left:** Example Attributions on the Standard, GridPG, and DiFull Settings. We show attributions for all methods on their typically evaluated layers, i.e. input for backpropagation-based and perturbation-based, and final layer for activation-based methods. Blue boxes denote the object bounding box (Standard) or the grid cell (GridPG, DiFull) respectively. For DiFull, we use images of the same class at the top-left and bottom-right corners as in our experiments. **Right:** Occlusion attributions for an example evaluated on GridPG, DiFull, and DiPart. The top-left and bottom-right corners contain two different species of dogs, which share similar low-level features, causing both to be attributed in GridPG. In contrast, our disconnected construction in DiFull and DiPart ensures that the bottom-right subimage does not influence the classification of the top-left, and thus should not be attributed by any attribution methods, even though some do erroneously.

Attribution Methods: We evaluate a diverse set of attribution methods, for an overview see Sec. 2. As discussed in Sec. 3.3, to apply those methods to intermediate network layers, we divide the full model into two virtual parts $\mathbf{f}_{\text{pre}}$ and $\mathbf{f}_{\text{explain}}$ and treat the output of $\mathbf{f}_{\text{pre}}$ as the input to $\mathbf{f}_{\text{explain}}$ to obtain importance attributions for those 'pre-processed' inputs. In particular, we evaluate the following methods. From the set of **backpropagation-based methods**, we evaluate on Guided Backpropagation [30], Gradient [27], IntGrad [32], and IxG [26]. From the set of **activation-based methods**, we evaluate on Grad-CAM [23], Grad-CAM++ [6], Ablation-CAM [8], Score-CAM [33], and Layer-CAM [12]. Note that in our framework, these methods can be regarded as using the classification head only (except [12]) for $\mathbf{f}_{\text{explain}}$, see Sec. 3.3. In order to evaluate them at earlier layers, we simply expand $\mathbf{f}_{\text{explain}}$ accordingly to include more network layers. From the set of **perturbation-based methods**, we evaluate Occlusion [34] and RISE [16]. These are typically evaluated on the input layer, and measure output changes when perturbing (occluding) the input (Fig. 3, left). Note that Occlusion involves sliding an occlusion kernel of size K with stride s over the input. We use $K{=}16, s{=}8$ for the input, and $K{=}5, s{=}2$ at the middle and final layers to account for the lower dimensionality of the feature maps. For RISE, we use $M{=}1000$ random masks , generated separately for evaluations at different network layers.

5. Experimental Results and Discussion

In this section, we first present the quantitative results for all attribution methods on GridPG, DiPart, and DiFull and compare their performance at multiple layers (5.1). Further, we present a simple smoothing mechanism that provides highly performant attributions on all three settings, and discuss architectural considerations that impact its effectiveness (5.2). Finally, we present qualitative results using AggAtt, and show its use in highlighting strengths and deficiencies of attribution methods (5.3).

5.1. Evaluation on GridPG, DiFull, and DiPart

We perform ML-Att evaluation using the input (Inp), a middle layer (Mid), and before the classification head (Fin) (x-ticks in Fig. 4) for all three quantitative evaluation settings (GridPG, DiFull, DiPart, minor columns in Fig. 4) discussed in Sec. 3. In the following, we discuss the methods' results, grouped by their 'method family': backpropagation-based, activation-based, and perturbation-based methods (major columns in Fig. 4).

Backpropagation-based methods: We observe that all methods perform poorly at the initial layer on GridPG (Fig. 4, left). Specifically, we observe gradient-based methods to yield noisy attributions that do not seem to reflect the grid structure of the images; i.e., positive attributions are nearly as likely to be found outside of a subimage for a specific class as they are to be found inside.

However, all methods improve on later layers. At the final layer, IntGrad and IxG show very good localization (comparable to Grad-CAM), which suggests that the methods may have similar explanatory power when compared on an equal footing. We note that IxG at the final layer has been previously proposed under the name DetGrad-CAM [20].

On DiFull, all methods show near-perfect localization across layers (Fig. 8). No attribution is given to disconnected subimages since the gradients with respect to them are zero (after all, they are *fully disconnected*); degradations for other layers can be attributed to the applied upsampling.

Similar results are seen in DiPart, but with decreasing localization when moving backwards from the classifier, which can be attributed to the fact that the receptive field can overlap with other subimages in this setting. Given the

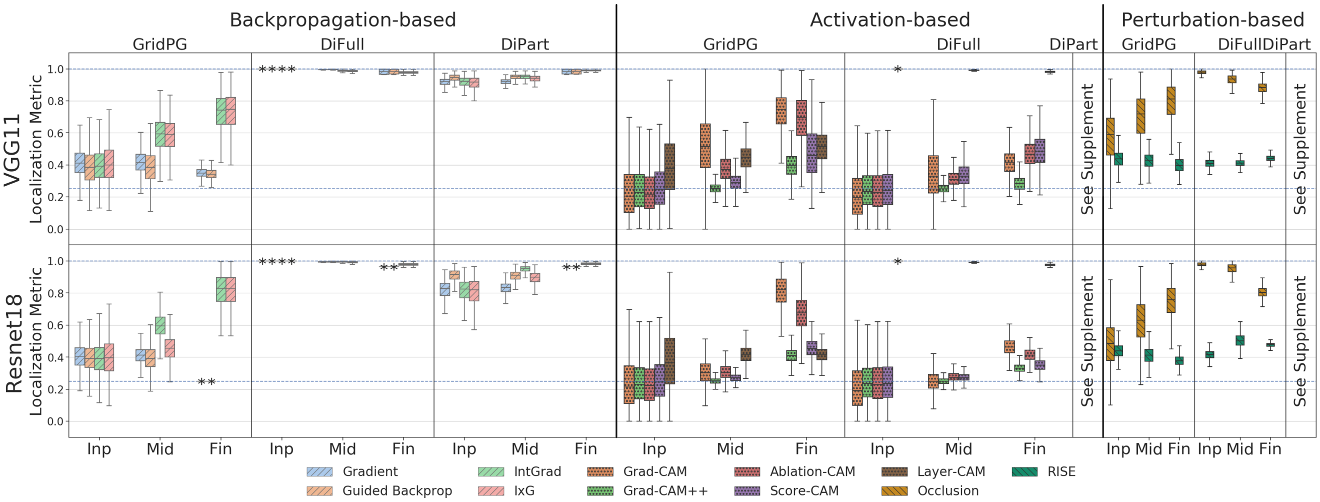


Fig. 4. **Quantitative Results on VGG11 and Resnet18.** For each metric, we evaluate all attribution methods with respect to the input image (Inp), a middle (Mid), and the final (Fin) spatial layer. We observe the performance to improve from *Inp* to *Fin* on most settings. Similar to the backpropagation-based methods, the results for methods on DiPart are very similar to those in DiFull for activation and perturbation-based; for details, see supplement. The symbol * denotes boxes that collapse to a single value, for better readability.

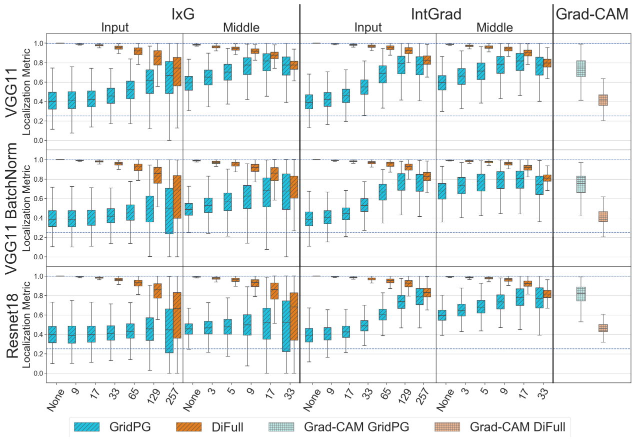


Fig. 5. **Smoothing the attributions** for IntGrad and IxG significantly improves their performance at the input image and middle layer. For reference, we show Grad-CAM on the *final* spatial layer.

similarity between the results for DiFull and DiPart, we restrict our discussion to DiFull; for DiPart, see supplement.

Activation-based methods: We see that all methods with the exception of Layer-CAM improve in localization performance from input to final layer on all three settings. Since attributions are computed using a scalar weighted sum of attribution maps, this improvement could be explained by improved localization of activations from later layers. In particular, localization is very poor at early layers, which is a well-known limitation (cf. [12]) of Grad-CAM. The weighting scheme also causes final layer attributions for all methods except Layer-CAM to perform worse on DiFull than on GridPG, since these methods attribute importance to both instances of the repeated class (Fig. 8). This issue is absent in Layer-CAM since it does not apply a pooling operation.

Perturbation-based methods: We observe (Fig. 4, right) Occlusion to perform well across layers on DiFull, since occluding disconnected subimages cannot affect the model outputs and are thus not attributed importance. However, the localization drops slightly for later layers. This is due to the fact that the relative size (w.r.t. activation map) of the overlap regions between occlusion kernels and adjacent subimages increases. This highlights the sensitivity of performance to the choice of hyperparameters, and the tradeoff between computational cost and performance.

On GridPG, Occlusion performance improves with layers. On the other hand, RISE performs poorly across all settings and layers. Since it uses random masks, pixels outside a target grid cell that share a mask with pixels within get attributed equally. So while attributions tend to concentrate more in the target grid cell, the performance can be inconsistent (Fig. 8).

5.2. Smoothing Attributions

From Sec. 5.1, we see that Grad-CAM localizes well at the final layer in GridPG, but performs poorly on all the other settings as a consequence of global pooling of gradients (for DiFull) and poor localization of early layer features (for GridPG early layers). Since IxG, in contrast, does not use a pooling operation, it performs well on DiFull at all layers and on GridPG at the final layer. However, it performs poorly at the input and middle layers on GridPG due to the noisiness of gradients; IntGrad shows similar results.

Devising an approach to eliminate this noise would provide an attribution method that performs well across settings and layers. Previous approaches to reduce noise include averaging attribution maps over many perturbed sam-

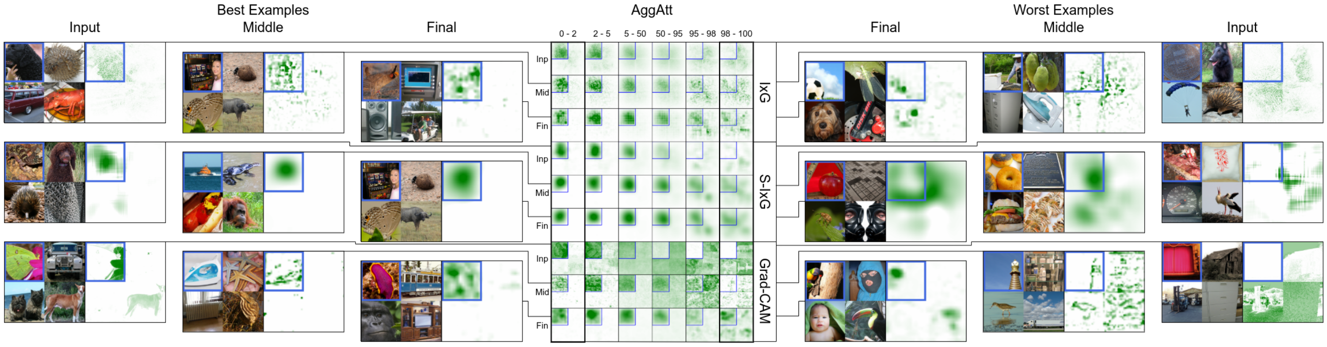


Fig. 6. **Qualitative Results for VGG11 on GridPG evaluated at the top-left corner**. *Centre:* Aggregate attributions sorted and binned in descending order of localization. Each column corresponds to a bin, and set of three rows corresponds to a method. For each method, the three rows from top to bottom show the aggregate attributions at the input, middle, and final spatial layers. *Left:* Examples from the first bin, which corresponds to the best set of attributions. *Right:* Similarly, we show examples from the last bin, which corresponds to the worst set of attributions. For smooth IxG, we use $K = 129$ for the input layer, $K = 17$ at the middle layer, and $K = 9$ at the final layer. All examples shown correspond to images whose attributions lie at the median position in their respective bins.

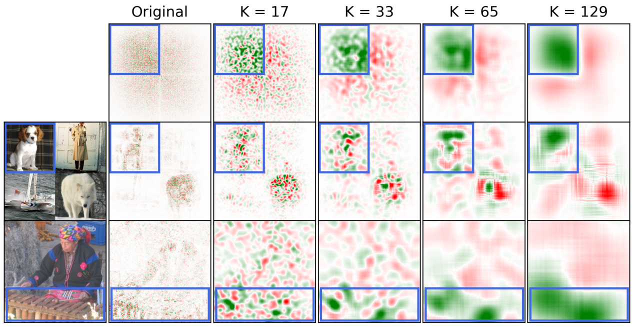


Fig. 7. **Qualitative Visualization of smoothing IxG attribution maps for various kernel sizes, including both positive and negative attributions.** *Top*: Aggregate attribution maps for VGG11 on GridPG at the top-left corner across the dataset. We see that positive attributions (green) aggregate to the top-left grid cell and negative attributions (red) aggregate outside when smoothing with large kernel sizes. *Middle and Bottom*: Examples of smoothing on a single grid and non-grid image. Positive attributions concentrate inside the bounding box when smoothed with large kernels.

ples (SmoothGrad [29], see supplement for a comparison) or adding a gradient penalty during training [13]. However, SmoothGrad is computationally expensive as it requires several passes on the network to obtain attributions, and is sensitive to the chosen perturbations. Similarly, adding a penalty term during training requires retraining the network.

Here, we propose to simply apply a Gaussian smoothing kernel on existing IntGrad and IxG attributions. We evaluate on DiFull and GridPG using several kernel sizes, using standard deviation $K/4$ for kernels of size K. We refer to the smooth versions as S-IntGrad and S-IxG respectively.

On VGG11 (Fig. 5, top), we find that S-IntGrad and S-IxG localize significantly better than IntGrad and IxG, and the performance improves with increasing kernel size. In detail, S-IntGrad *on the input layer* with $K{=}257$ outperforms Grad-CAM *on the final layer*, despite explaining the full network. While performance on DiFull drops slightly as smoothing leaks attributions across grid boundaries, both S-IntGrad and S-IxG localize well across settings and layers. However, on Resnet18 (Fig. 5, bottom), while S-IntGrad improves similarly, S-IxG does not, which we discuss next.

Impact of Network Architecture: A key difference between the VGG11 and Resnet18 architectures used in our experiments is that VGG11 does not have batch normalization (BatchNorm) layers. We note that batch norm effectively randomizes the sign of the input vectors to the subsequent layer, by centering those inputs around the origin (cf. [11,13]). Since the sign of the input determines whether a contribution (weighted input) is *positive* or *negative*, a BatchNorm layer will randomize the sign of the contribution and the 'valence' of the contributions will be encoded in the BatchNorm biases. To test our hypothesis, we evaluate S-IxG on a VGG11 with BatchNorm layers (Fig. 5, middle), and observe results similar to Resnet18: i.e., we observe no systematic improvement by increasing the kernel size of the Gaussian smoothing operation. This shows that the architectural choices of a model can have a significant impact on the performance of attribution methods.

5.3. Qualitative Evaluation using AggAtt

In this section, we present qualitative results using AggAtt for select attributions evaluated on GridPG and DiFull and multiple layers. First, to investigate the qualitative impact of smoothing, we use AggAtt to compare IxG, S-IxG, and Grad-CAM attributions on GridPG on multiple layers. We employ AggAtt on DiFull to highlight specific characteristics and failure cases of some attribution methods.

AggAtt on GridPG: We show AggAtt results for IxG, S-IxG, and Grad-CAM at three layers on GridPG using VGG11 on the images at the top-left corner (Fig. 6). For each method, a set of three rows corresponds to the attributions at input, middle, and final layers. For S-IxG, we set K to 129, 17, and 9 respectively. We further show individual

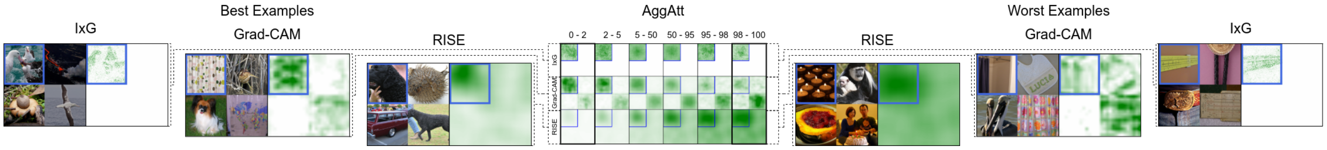


Fig. 8. **Qualitative Results for VGG11 on DiFull evaluated at the top-left corner**. *Centre:* Aggregate attributions sorted and binned in descending order of localization. Each column corresponds to a bin and each row corresponds to a method applied at its standard layer. *Left:* Examples from the first bin, which corresponds to the best set of attributions. *Right:* Examples from the last bin, which corresponds to the worst set of attributions. All examples shown correspond to images whose attributions lie at the median position in their bins.

samples (median bin) of the first and last bins per method.

We observe that the aggregate visualizations are consistent with the quantitative results (Figs. 4 and 5) and the individual examples shown for each bin. The performance improves for IxG and Grad-CAM from input to final layer, while S-IxG localizes well across three layers. Finally, the last two columns show that all the attribution methods perform 'poorly' for some inputs; e.g., we find that IxG and Grad-CAM on the final layer attribute importance to other subimages if they exhibit features that are consistent with the class in the top-left subimage. While the attributions might be conceived as incorrect, we find that many 'failure cases' on GridPG highlight features that the underlying model might in fact use, even if they are in another subimage. Given the lack of ground truth, it is difficult to assess whether these attributions faithfully reflect model behaviour or deficiencies of the attribution methods.

Despite explaining significantly more layers, S-IntGrad and S-IxG at the input layer not only match Grad-CAM at the final layer quantitatively (Fig. 5) and qualitatively (Fig. 6), but are also highly consistent with it for individual explanations. Specifically, the Spearman rank correlation between the localization scores of Grad-CAM (final layer) and S-IntGrad (input layer) increases significantly as compared to IntGrad (input layer) (e.g., $0.34 \rightarrow 0.80$ on VGG11), implying that their attributions for any input tend to lie in the same AggAtt bins (see supplement).

To further understand the effect of smoothing, we visualize S-IxG with varying kernel sizes while including negative attributions (Fig. 7). The top row shows aggregate attributions across the dataset, while the middle and bottom rows show an example under the GridPG and standard localization settings respectively. We observe that while IxG attributions appear noisy (column 2), smoothing causes positive and negative attributions to cleanly separate out, with the positive attributions concentrating around the object. For instance, in the second row, IxG attributions concentrate around both the dog and the wolf, but S-IxG with $K = 129$ correctly attributes only the dog positively. This could indicate a limited effective receptive field (RF) [15] of the models. Specifically, note that for piece-wise linear models, summing the contributions (given by IxG) over all input dimensions within the RF exactly yields the output logit (disregarding biases). Models with a small RF would thus be well summarised by S-IxG for an adequately sized kernel; we elaborate on this in the supplement.

AggAtt on DiFull: We visually evaluate attributions on DiFull for one method per method family, i.e., from backpropagation-based (IxG, input layer), activation-based (Grad-CAM, final layer), and perturbation-based (RISE, input layer) methods at their standard layers (Fig. 8). The top row corroborates the near-perfect localization shown by the backpropagation-based methods on DiFull. The middle row shows that Grad-CAM attributions concentrate at the top-left and bottom-right corners, which contain images of the same class, since global pooling of gradients makes it unable to distinguish between the two even though only the top-left instance (here) influences classification. Finally, for RISE, we observe that while attributions localize well for around half the images, the use of random masks results in noisy attributions for the bottom half.

6. Conclusion

In this work, we proposed schemes to evaluate model-faithfulness of attribution methods in a fair and systematic manner. We first proposed a quantitative metric, DiFull, that constrains input regions that can affect classification. This yields regions in which any attribution is necessarily unfaithful and allows to highlight distinct failure cases of some attribution methods. Then, we proposed a multi-layer evaluation scheme, ML-Att, that compares methods fairly, and found that the performance gap between methods narrows considerably when done so. Finally, we proposed AggAtt, a novel qualitative evaluation scheme that allows for succinctly visualizing the variation in performance of attribution methods in a systematic and holistic manner. Overall, we find that fair comparisons, holistic evaluations (DiFull, GridPG, AggAtt, ML-Att), and careful disentanglement of model behaviour from the explanations provide better insights in the performance of attribution methods.

Limitations. We note that while our method can distinguish between possibly correct and impossibly correct attributions, our method cannot evaluate the correctness of specific attributions within the target grid cells, since ground truths for these are unknown.

References

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity Checks for Saliency Maps. In *NeurIPS*, 2018. 2

[2] Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. Towards better understanding of gradient-based attribution methods for Deep Neural Networks. In *ICLR*, 2018. 2

[3] Anna Arias-Duart, Ferran Parés, and Dario Garcia-Gasulla. Who Explains the Explanation? Quantitatively Assessing Feature Attribution Methods. *arXiv preprint arXiv:2109.15035*, 2021. 2, 4

[4] Moritz Böhle, Mario Fritz, and Bernt Schiele. Convolutional Dynamic Alignment Networks for Interpretable Classifications. In *CVPR*, pages 10029–10038, 2021. 2, 3, 4

[5] Chunshui Cao, Xianming Liu, Yi Yang, Yinan Yu, Jiang Wang, Zilei Wang, Yongzhen Huang, Liang Wang, Chang Huang, Wei Xu, et al. Look and Think Twice: Capturing Top-Down Visual Attention with Feedback Convolutional Neural Networks. In *ICCV*, pages 2956–2964, 2015. 2

[6] Aditya Chattopadhyay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-CAM++: Improved Visual Explanations for Deep Convolutional Networks. In *WACV*, pages 839–847, 2018. 2, 5

[7] Piotr Dabkowski and Yarin Gal. Real Time Image Saliency for Black Box Classifiers. In *NeurIPS*, 2017. 2

[8] Saurabh Desai and Harish G. Ramaswamy. Ablation-CAM: Visual Explanations for Deep Convolutional Network via Gradient-free Localization. In *WACV*, pages 983–991, 2020. 2, 5

[9] Ruth C Fong and Andrea Vedaldi. Interpretable Explanations of Black Boxes by Meaningful Perturbation. In *ICCV*, pages 3429–3437, 2017. 2

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, pages 770–778, 2016. 4

[11] Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *ICML*, pages 448–456, 2015. 7

[12] Peng-Tao Jiang, Chang-Bin Zhang, Qibin Hou, Ming-Ming Cheng, and Yunchao Wei. LayerCAM: Exploring Hierarchical Class Activation Maps for Localization. *IEEE TIP*, 30:5875–5888, 2021. 2, 5, 6

[13] Keisuke Kiritoshi, Ryosuke Tanno, and Tomonori Izumitani. L1-Norm Gradient Penalty for Noise Reduction of Attribution Maps. In *CVPRW*, pages 118–121, 2019. 7

[14] Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images. 2009. 4

[15] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the Effective Receptive Field in Deep Convolutional Neural Networks. In *NeurIPS*, 2016. 8

[16] Vitali Petsiuk, Abir Das, and Kate Saenko. RISE: Randomized Input Sampling for Explanation of Black-box Models. In *BMVC*, 2018. 2, 5

[17] Sylvestre-Alvise Rebuffi, Ruth Fong, Xu Ji, and Andrea Vedaldi. There and Back Again: Revisiting Backpropagation Saliency Methods. In *CVPR*, pages 8839–8848, 2020. 2

[18] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *KDD*, pages 1135–1144, 2016. 2

[19] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 115(3):211–252, 2015. 4

[20] Aniruddha Saha, Akshayvarun Subramanya, Koninika Patil, and Hamed Pirsiavash. Role of Spatial Context in Adversarial Robustness for Object Detection. In *CVPRW*, pages 784–785, 2020. 5

[21] Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. Evaluating the visualization of what a Deep Neural Network has learned. *IEEE Trans. Neural Netw. Learn. Syst.*, 28(11):2660–2673, 2016. 2

[22] Karl Schulz, Leon Sixt, Federico Tombari, and Tim Landgraf. Restricting the Flow: Information Bottlenecks for Attribution. In *ICLR*, 2020. 2

[23] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *ICCV*, pages 618–626, 2017. 2, 4, 5

[24] Harshay Shah, Prateek Jain, and Praneeth Netrapalli. Do Input Gradients Highlight Discriminative Features? In *NeurIPS*, 2021. 2

[25] Rakshith Shetty, Bernt Schiele, and Mario Fritz. Not Using the Car to See the Sidewalk - Quantifying and Controlling the Effects of Context in Classification and Segmentation. In *CVPR*, pages 8218–8226, 2019. 2

[26] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning Important Features Through Propagating Activation Differences. In *ICML*, pages 3145–3153, 2017. 2, 5

[27] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. In *ICLRW*, 2014. 2, 5

[28] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *ICLR*, 2015. 4

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

[30] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for Simplicity: The All Convolutional Net. In *ICLRW*, 2015. 2, 5

[31] Suraj Srinivas and François Fleuret. Full-Gradient Representation for Neural Network Visualization. In *NeurIPS*, 2019. 2

[32] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic Attribution for Deep Networks. In *ICML*, pages 3319–3328, 2017. 2, 5

[33] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-CAM: Score-Weighted Visual Explanations for Convolutional Neural Networks. In *CVPRW*, pages 111–119, 2020. 2, 5

[34] Matthew D Zeiler and Rob Fergus. Visualizing and Understanding Convolutional Networks. In *ECCV*, pages 818–833, 2014. 2, 5

[35] Jianming Zhang, Sarah Adel Bargal, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-Down Neural Attention by Excitation Backprop. *IJCV*, 126(10):1084–1102, 2018. 2

[36] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning Deep Features for Discriminative Localization. In *CVPR*, pages 2921–2929, 2016. 2