Towards Better Understanding Attribution Methods
Supplementary Material

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

Contents

In this supplement, we provide additional details and results of our evaluation on our proposed settings. First, we provide additional quantitative results on ImageNet [S10] (Sec. A), including complete results on DiPart, followed by complete qualitative results using AggAtt (Sec. B) with examples. Then, we investigate the correlation between the localization scores of attribution methods (Sec. C) and compare it with the trends seen from the qualitative and quantitative results. This is followed by a discussion of the impact of smoothing attributions (Sec. D), including a comparison of the similarity of Grad-CAM [S11] with S-IntGrad and S-IxG on the same set of examples across AggAtt bins. In Sec. E, we provide quantitative results at all spatial layers for both models, and show that it reflects the trends observed from the three chosen layers in our experiments. This is followed by an analysis of the computational costs of each evaluation setting (Sec. F), and a comparison of our proposed smoothing technique with SmoothGrad [S15] (Sec. G). Then, we describe further implementation details (Sec. H) of our experiments. Finally, we provide results of our approach on CIFAR10 [S6] (Sec. I), and find similar trends as on ImageNet across our settings.

A. Quantitative Results Including DiPart

We provide the full results of our quantitative evaluation on the Grid Pointing Game [S1] (GridPG), DiFull, and DiPart using the backpropagation-based (Fig. 9, top), activation-based (Fig. 9, middle), and perturbation-based (Fig. 9, bottom) methods on VGG11 [S14] (Fig. 9, left) and Resnet18 [S2] (Fig. 9, right).

It can be seen that the performance on DiFull and DiPart is very similar across all three evaluation settings and the three layers. The most significant difference between the two can be seen among the backpropagation-based methods and Layer-CAM [S4] (Fig. 9, row 1, cols. 2-3,5-6). On DiFull, these methods show near-perfect localization, since the gradients of the outputs from each classification head that are used to assign importance are zero with respect to weights and activations of all grid cells disconnected from that head. On the other hand, the receptive field of the convolutional layers can overlap adjacent grid cells in DiPart, and the gradients of the outputs from the classification heads can thus have non-zero values with respect to inputs and activations from these adjacent grid regions. This also results in decreasing localization scores when moving backwards from the classifier.

Furthermore, the localization scores for Gradient [S13] and Guided Backprop [S16] are constant at the final layer for Resnet18 (Fig. 9, row 1, cols. 4-6). This is because this layer is immediately followed by a global average pooling layer, due to which all activations at this layer get an equal share of the gradients.

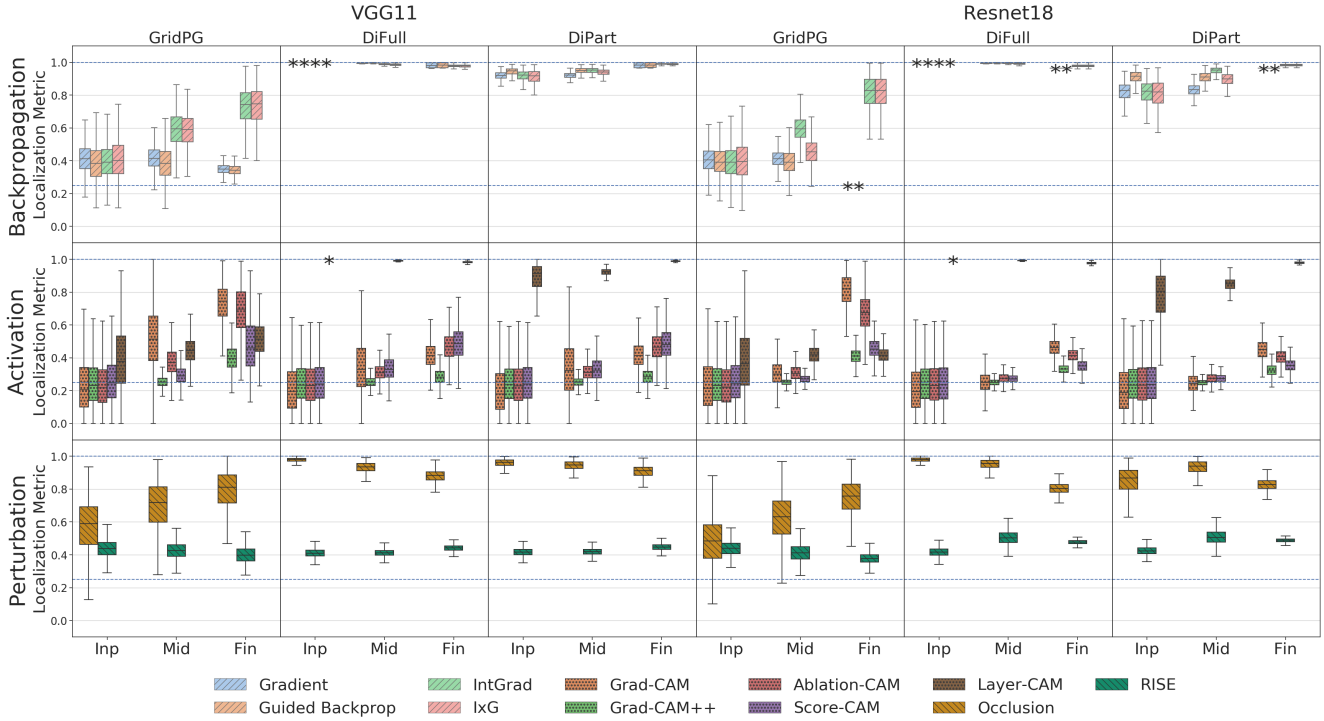


Figure 9. **Quantitative Results on ImageNet**. We evaluate the localization scores each attribution method at the input (Inp), middle (Mid), and final (Fin) convolutional layers, on each of GridPG, DiFull, and DiPart using VGG11 (left) and Resnet18 (right). *Top:* Backpropagation-based methods. *Middle:* Activation-based methods. *Bottom:* Perturbation-based methods. The two horizontal dotted lines mark localization scores of 1.0 and 0.25, which correspond to perfect and random localization, respectively. We use the "*" symbol to show boxes that collapse to a single point, for better readability.

B. Qualitative Results using AggAtt

In this section, we present additional qualitative results using our AggAtt evaluation along with examples of attributions from each bin, for each of GridPG [S1] (Sec. B.1), DiFull (Sec. B.2), and DiPart (Sec. B.3).

B.1. GridPG

Fig. 10 and Fig. 11 show examples from the median position of each AggAtt bin for each attribution method at the input and final layers, respectively, evaluated on GridPG at the top-left grid cell using VGG11 [S14]. At the input layer (Fig. 10), we observe that the **backpropagation-based methods** show noisy attributions that do not strongly localize to the top-left grid cell. This corroborates the poor quantitative performance of these methods at the input layer (Fig. 9, top). With the exception of Layer-CAM [S4], the **activation-based methods**, on the other hand, show strong attributions across all four grid cells, and localize very poorly. They appear to highlight the edges across the input irrespective of the class of each grid cell. This also agrees with the quantitative results (Fig. 9, middle), where the median localization score of these methods is below the uniform attribution baseline. Layer-CAM, being similar to IxG [S12], lies at the interface between activation and backpropagation-based methods, and also shows weak and noisy attributions. The **perturbation-based methods** visually show a high variance in attributions. While they localize well for about half the dataset (first three bins), the bottom half (last three bins) shows noisy and poorly localized attributions, which again agrees with the quantitative results (Fig. 9, bottom). This further shows how evaluating on individual inputs can be misleading, and the utility of AggAtt for obtaining a holistic view across the dataset.

Figure 10. **Examples from each AggAtt bin for each method at the input layer on GridPG using VGG11.** From each bin, the image and its attribution at the median position are shown.

At the final layer (Fig. 11), attributions from Gradient [S13] and Guided Backprop [S16] are very noisy and only slightly concentrate at the top-left cell. The checkerboard-like pattern is a consequence of the max pooling operation after the final layer, which allocates all the gradient only to the maximum activation. Gradients from each position of the sliding classification kernel then get averaged to form the attributions. The localization of IntGrad [S17], IxG, Grad-CAM [S11], and Occlusion [S18] improve considerably as compared to the input layer, which agrees with the quantitative results, and shows that diverse methods can show similar performance when compared fairly. The performance of the other activation-based methods and RISE [S9] improves to some extent, but is still poorly localized for around the half the dataset.

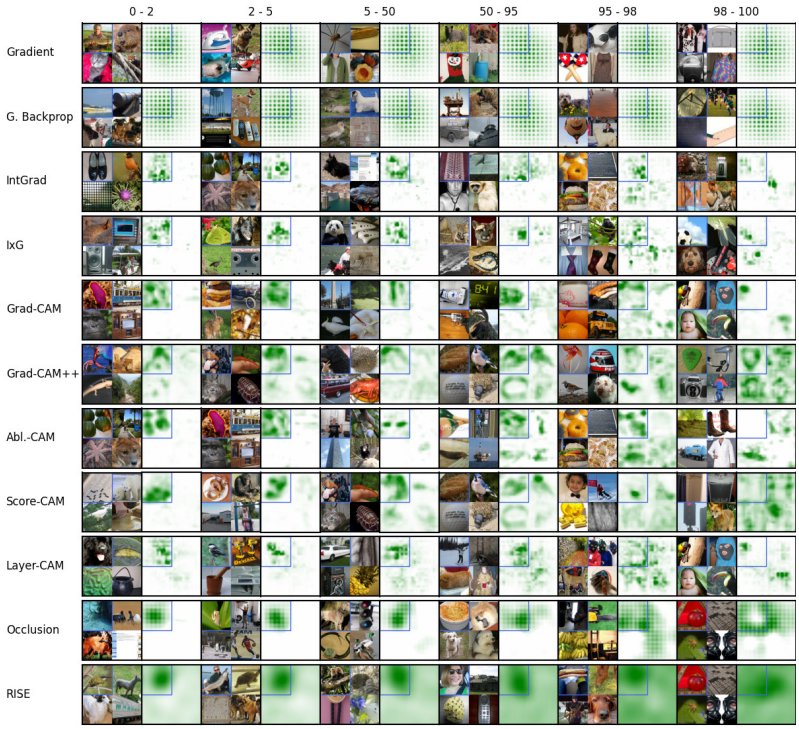


Figure 11. **Examples from each AggAtt bin for each method at the final layer on GridPG using VGG11.** From each bin, the image and its attribution at the median position are shown.

Finally, we show the AggAtt bins for all methods at all three layers using both VGG11 and Resnet18 [S2] in Fig. 12. We see that the AggAtt bins reflect the trends observed in the examples in each bin, and serve as a useful tool for visualization.

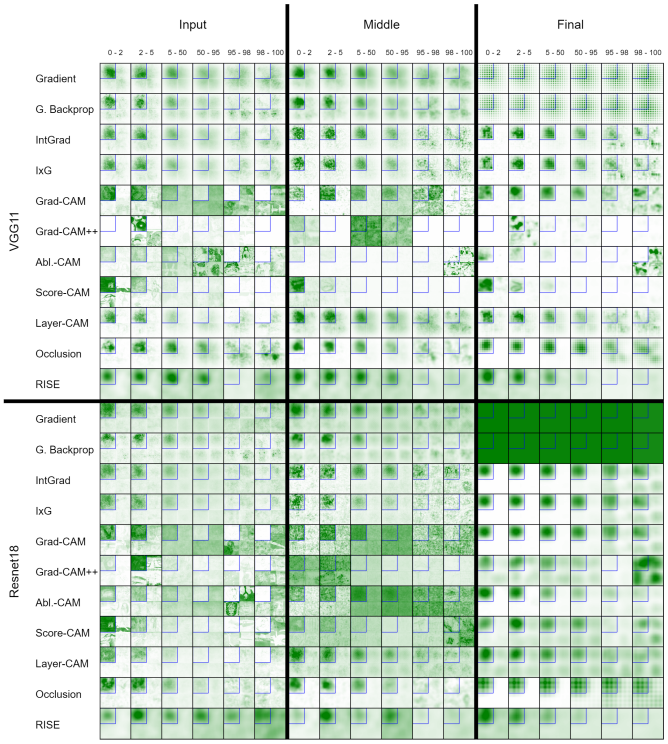


Figure 12. **AggAtt Evaluation on GridPG for all methods at the input, middle, and final layers using VGG11 and Resnet18.**

B.2. DiFull

Fig. 13 and Fig. 14 show examples from the median position of each AggAtt bin for each attribution method at the input and final layers, respectively, evaluated on DiFull at the top-left grid cell using VGG11. At the input layer (Fig. 13), the backpropagation-based methods and Layer-CAM show perfect localization across the dataset. This is explained by the disconnected construction of DiFull, and agrees with the quantitative results shown in Fig. 9). The activation-based methods show very poor localization that appear visually similar to the attributions observed on GridPG (Sec. B.1). Occlusion shows near-perfect localization, since the placement of the classification kernel at any location not overlapping with the top-left grid cell does not influence the output in the DiFull setting. RISE still produces noisy attributions across the dataset. While only the top-left grid cell influences the output, the use of random masks causes input regions that share masks with inputs in the top-left cell to also get attributed.

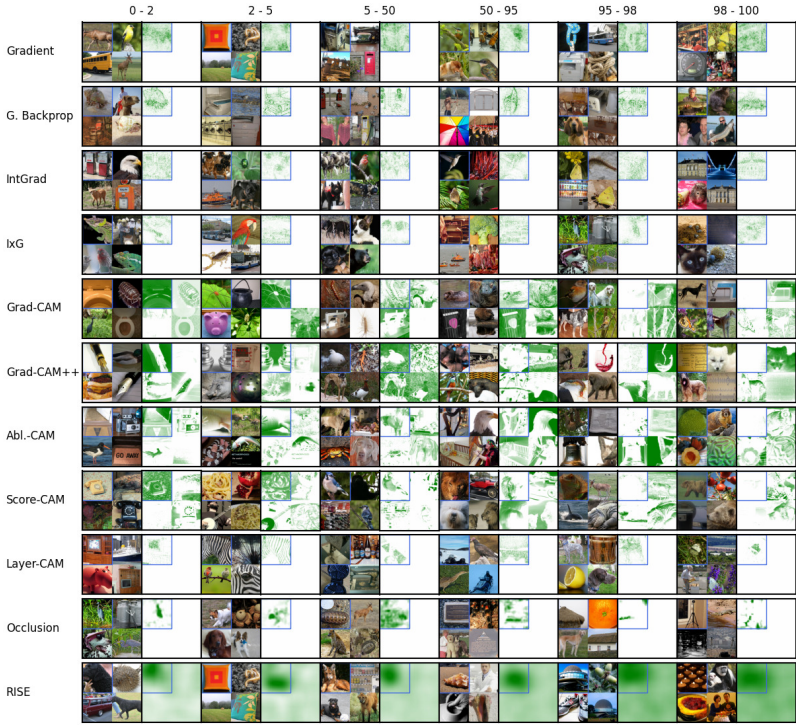


Figure 13. **Examples from each AggAtt bin for each method at the input layer on DiFull using VGG11.** From each bin, the image and its attribution at the median position are shown.

At the final layer (Fig. 14), the backpropagation-based methods and Layer-CAM still show perfect localization, for the same reason as discussed above. Attributions from Gradient and Guided Backprop show similar artifacts as seen with GridPG (Sec. B.1), but are localized to the top-left cell. The activation-based methods apart from Layer-CAM concentrate their attributions at the top-left and bottom-right grid cells, particularly in the early bins. This is because both these cells contain images from the same class, and the weighing of activation maps by these methods using a scalar value causes both to be attributed, even though only the instance at the top-left influences the classification. Further, Occlusion and RISE show similar results as at the input layer. The attributions of Occlusion are noticeably lower in resolution, since the relative size of the occlusion kernel as compared to the activation map is much larger at the final layer.

Figure 14. **Examples from each AggAtt bin for each method at the final layer on DiFull using VGG11.** From each bin, the image and its attribution at the median position are shown.

Finally, we show the AggAtt bins for all methods at all three layers using both VGG11 and Resnet18 in Fig. 15, and see that they reflect the trends observed in the individual examples seen from each bin.

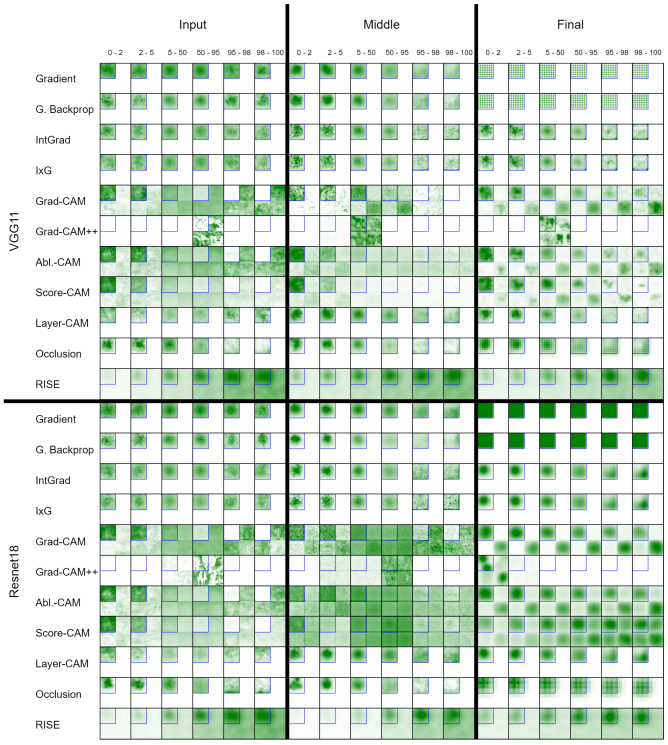


Figure 15. **AggAtt Evaluation on DiFull for all methods at the input, middle, and final layers using VGG11 and Resnet18.**

B.3. DiPart

Fig. 16 and Fig. 17 show examples from the median position of each AggAtt bin for each attribution method at the input and final layers, respectively, evaluated on DiPart at the top-left grid cell using VGG11. In addition, Fig. 18 shows the AggAtt bins for all methods at all three layers using both VGG11 and Resnet18. As observed with the quantitative results (Sec. A, the performance seen visually on DiPart across the three layers is very similar to that on DiFull (Sec. B.2). However, they slightly differ in the case of the backpropagation-based methods and Layer-CAM, particularly at the input layer (Fig. 16). This is because unlike in DiFull, the grid cells are only partially disconnected, and the receptive field of the convolutional layers can overlap adjacent grid cells to some extent. Nevertheless, as can be seen here, only a small boundary region around the top-left grid cell receives attributions, and the difference is not visually very perceivable. This further shows that the DiPart setting can be thought of as a natural extension for DiFull, that mostly shares the requisite property without being an entirely constructed setting.

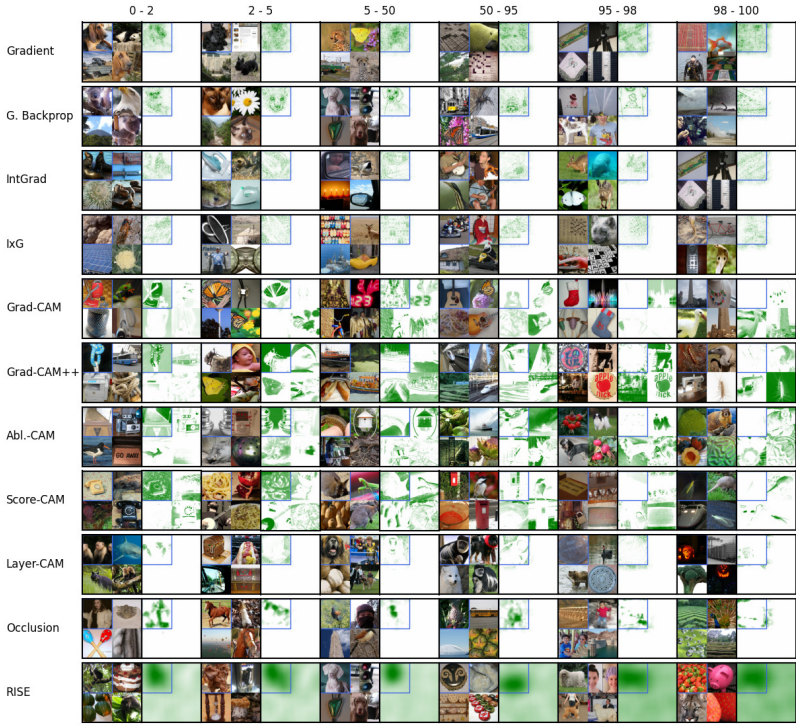


Figure 16. **Examples from each AggAtt bin for each method at the input layer on DiPart using VGG11.** From each bin, the image and its attribution at the median position are shown.

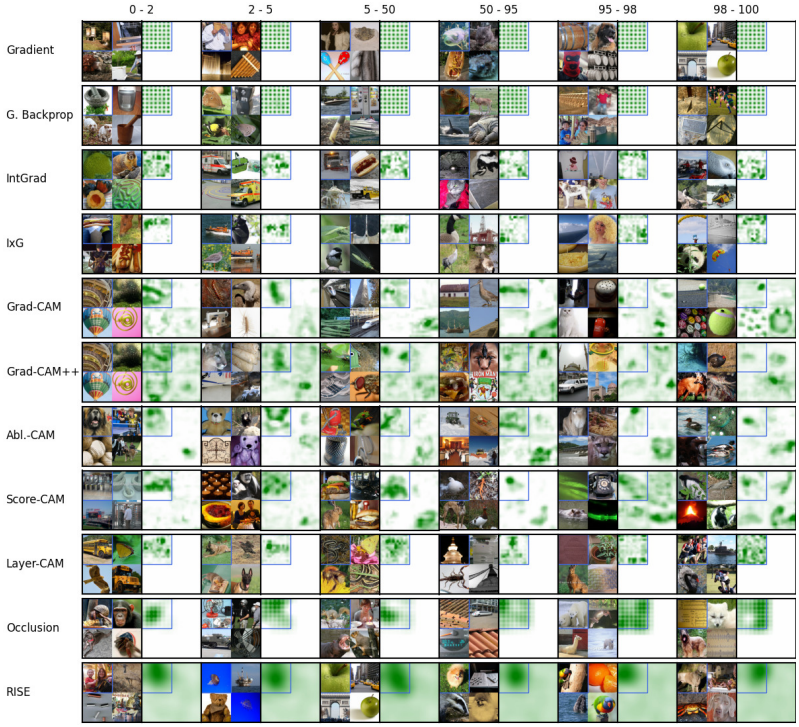


Figure 17. **Examples from each AggAtt bin for each method at the final layer on DiPart using VGG11.** From each bin, the image and its attribution at the median position are shown.

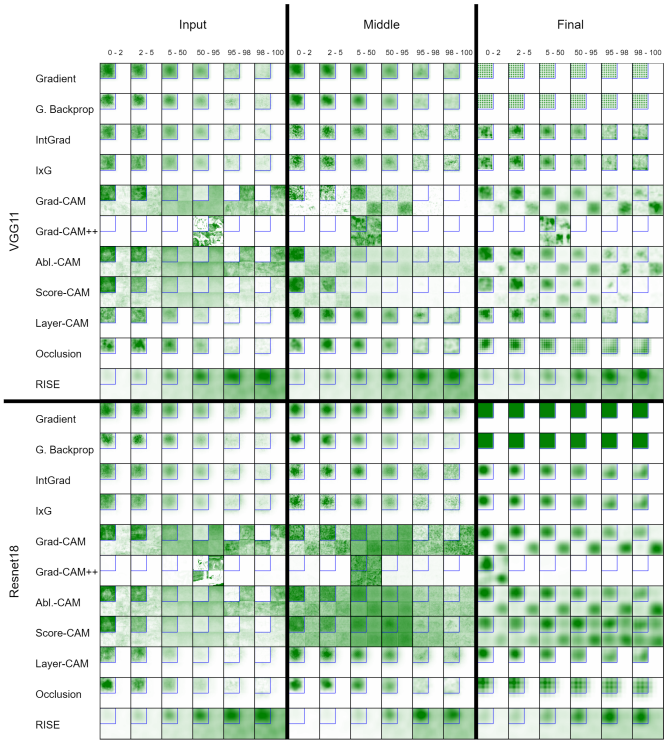


Figure 18. **AggAtt Evaluation on DiPart for all methods at the input, middle, and final layers using VGG11 and Resnet18.**

C. Correlation between Attributions

From the quantitative (Fig. 9) and qualitative (Fig. 12) results, we observed that diverse methods perform similarly on GridPG [S1] both in terms of localization score and through AggAtt visualizations when evaluated fairly. This was particularly the case with IntGrad [S17], IxG [S12], Grad-CAM [S11], and Occlusion [S18], when evaluated at the final layer. We also found (Sec 5.2 in the paper) that smoothing IntGrad and IxG (the result of which we call S-IntGrad and S-IxG) attributions evaluated at the *input layer* leads to visually and quantitatively similar performance as Grad-CAM evaluated at the *final layer*. In this section, we investigate this further, and study the correlation of these methods at the level of individual attributions. In particular, we compute the Spearman rank correlation coefficient of the localization scores using VGG11 [S14] of every pair of methods from each of the three layers. The results are shown in Fig. 19.

We observe that at the input layer (Fig. 19, top-left corner), the activation-based methods are poorly correlated with each other and with the backpropagation and perturbation-based methods. This also agrees with the poor localization of these methods seen previously (Fig. 9, Fig. 10). The backpropagation-based and perturbation-based methods, on the other hand, show moderate to strong correlation amongst and with each other. Similar results can be seen when comparing methods at the middle layer with the input layer and the final layer (Fig. 19, edge centres). However, when compared at the middle layer (Fig. 19, middle), the activation-based methods still correlate poorly with other methods, but the strength of the correlation improves in general.

Further, when compared at the final layer (Fig. 19, bottom-right corner), all methods show moderate to strong correlations with each other. This could be because generating explanations at the final layer is a significantly easier task as compared to doing so at the input, since the activations are used as is and only the classification layers' outputs are explained. The pairs with very strong positive correlation also show that attribution methods with diverse mechanisms can perform similarly when evaluated fairly. Finally, we observe that the activation-based methods at the final layer, instead of the input layer, correlate much better with the other methods at the input layer (Fig. 19, top-right, bottom-left).

We also observe that S-IntGrad and S-IxG at the input layer correlate well with the best-performing methods (IntGrad, IxG, Grad-CAM, Occlusion) at the final layer. Further, this marks a significant improvement when compared with IntGrad and IxG at the input layer. For example, IntGrad at the input layer compared with Grad-CAM at the final layer results in a correlation coefficient of 0.34, while S-IntGrad results in a correlation coefficient of 0.80.

We further study the effect of smoothing in Tabs. 1 and 2. We observe that the correlation between S-IntGrad and S-IxG improves significantly over IntGrad and IxG for VGG11 when using large kernels. However, for Resnet18 [S2], the improvement for S-IxG is very small. This agrees with the quantitative localization performance of these methods (Sec 5.2 in the paper). This shows that beyond aggregate visual similarity and quantitative performance, smoothing IntGrad and IxG can produce explanations at the input layer that are individually similar to Grad-CAM at the final layer, while also explaining the full network and performing significantly better on DiFull. We further visually compare the impact of smoothing in Sec. D.

	Original	$K = 9$	$K = 17$	$K = 33$	$K = 65$	$K = 129$	$K = 257$
VGG11	0.34	0.42	0.52	0.69	0.78	**0.80**	0.71
Resnet18	0.18	0.21	0.27	0.40	0.55	**0.63**	0.61

Table 1. **Spearman rank correlation coefficients between Grad-CAM at the final layer and S-IntGrad at the input layer on GridPG for varying degrees of smoothing.** The first column shows the correlation with the original unsmoothed version. We observe that the correlation improves significantly for both VGG11 and Resnet18 when smoothing with large kernel sizes.

	Original	$K = 9$	$K = 17$	$K = 33$	$K = 65$	$K = 129$	$K = 257$
VGG11	0.27	0.28	0.33	0.43	**0.49**	0.44	0.34
Resnet18	0.14	0.13	0.15	0.17	**0.18**	0.13	0.05

Table 2. **Spearman rank correlation coefficients between Grad-CAM at the final layer and S-IxG at the input layer on GridPG for varying degrees of smoothing.** The first column shows the correlation with the original unsmoothed version. We observe that the correlation improves for VGG11, but does not significantly improve for Resnet18.

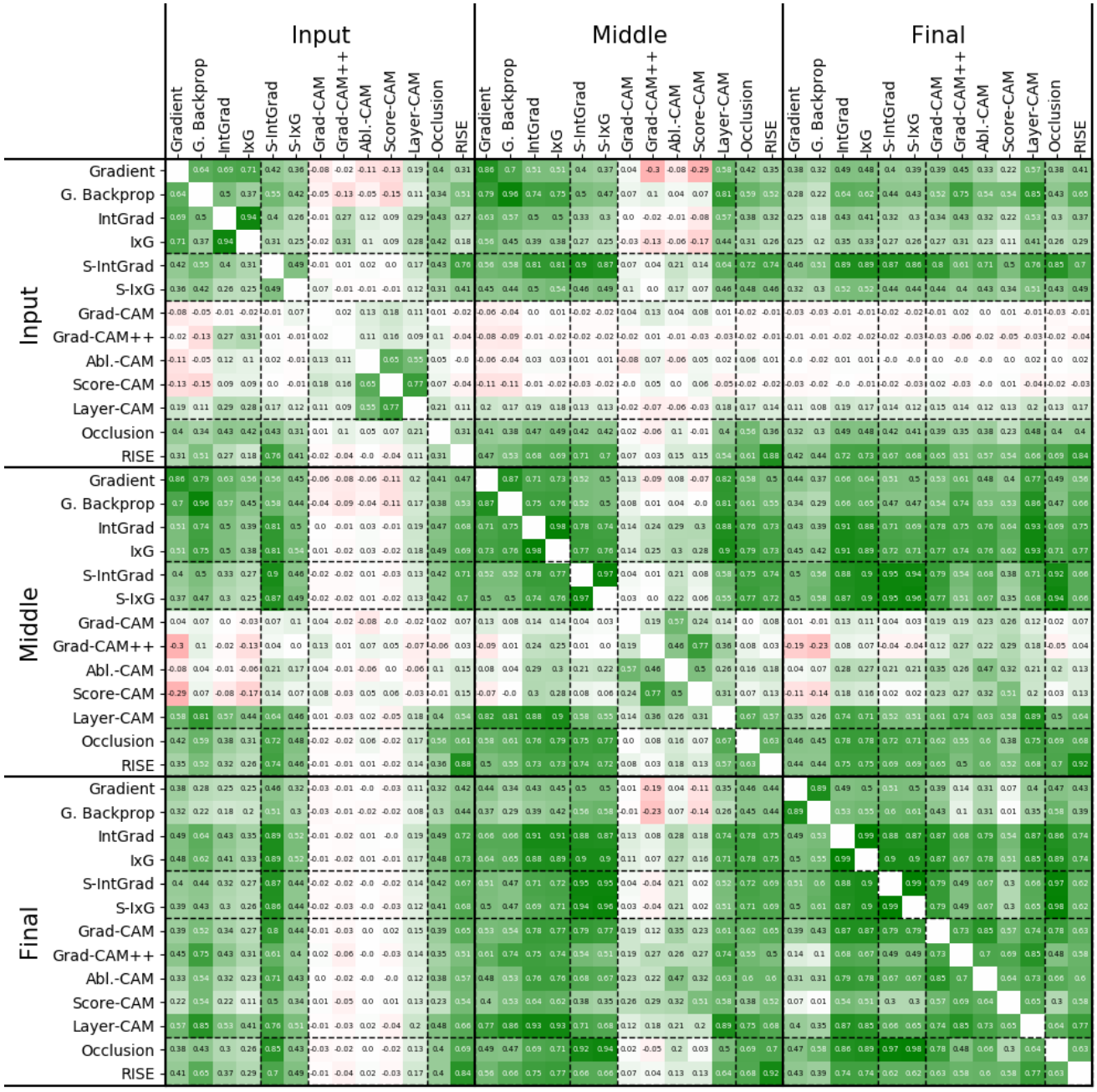


Figure 19. **Spearman rank correlation coefficients between each pair of methods at all three layers on GridPG for VGG11.** We observe that diverse methods (IntGrad, IxG, Grad-CAM, Occlusion) correlate strongly when evaluated at the final layer (bottom-right), which agrees with their similar quantitative and AggAtt performance. Further, S-IntGrad and S-IxG at the input layer correlate significantly stronger with Grad-CAM at the final layer as compared to IntGrad and IxG at the input layers. This, combined with the similarity in quantitative and AggAtt performance, shows the utility of smoothing in obtaining attributions that localize well while also performing much better in not attributing impossible regions specified by DiFull.

D. Impact of Smoothing Attributions

In this section, we explore the impact of smoothing attributions. First, we briefly discuss a possible reason for the improvement in localization of attributions after smoothing (Sec. D.1. Then, we visualize the impact of smoothing through examples and AggAtt visualizations (Sec. D.2). Further, we also compare the performance of Grad-CAM [S11] at the final layer with S-IntGrad and S-IxG at the input layer across the same examples from each bin and show their similarities across bins (Sec. D.3).

D.1. Effect of Smoothing

We believe that our smoothing results highlight an interesting aspect of piece-wise linear models (PLMs), which goes beyond mere practical improvements. For PLMs (such as the models used here), IxG [S12] yields the exact pixel contributions according to the linear mapping given by the PLM. In other words, the sum of IxG attributions over all pixels yields exactly (ignoring biases) the model output. If the effective receptive field of the model is small (cf. [S7]), sum pooling IxG with a kernel of the same size accurately computes the model's local output (apart from the influence of bias terms). Our method of smoothing IxG with a Gaussian kernel performs a weighted average pooling of attributions in the local region around each pixel, which produces a similar effect and appears to summarize the effect of the pixels in the local region to the model's output, which leads to less noisy attributions and better localization.

D.2. AggAtt Evaluation after Smoothing

In Fig. 20 and Fig. 21, we show examples from each AggAtt bin for S-IntGrad and S-IxG at the input layer for two different kernel sizes, and compare with IntGrad [S17] and IxG at the input layer respectively. We observe that the localization performance significantly improves with increasing kernel size, and produces much stronger attributions for the target grid cell. In Fig. 22, we show the AggAtt bins for these methods on both VGG11 [S14] and Resnet18 [S2]. We see that this reflects the trends seen from the examples, and also clearly shows the relative ineffectiveness of smoothing IxG for Resnet18 (Fig. 22 bottom right and Tab. 2).

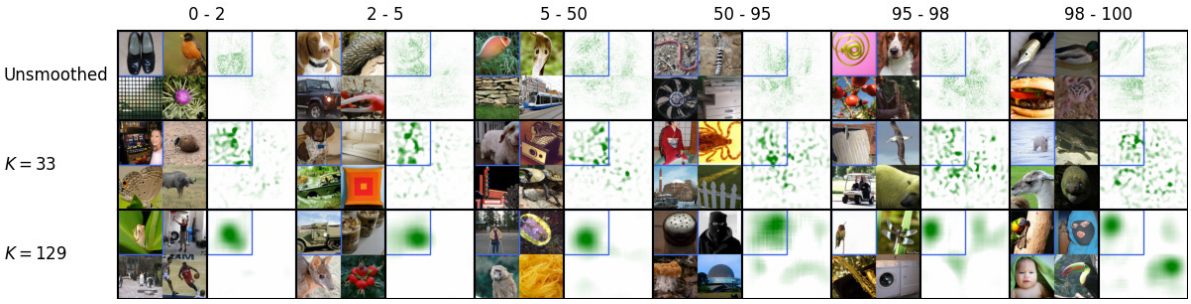


Figure 20. **Examples from each AggAtt bin after smoothing IntGrad attributions on GridPG using VGG11.** From each bin, the image and its attribution at the median position are shown.

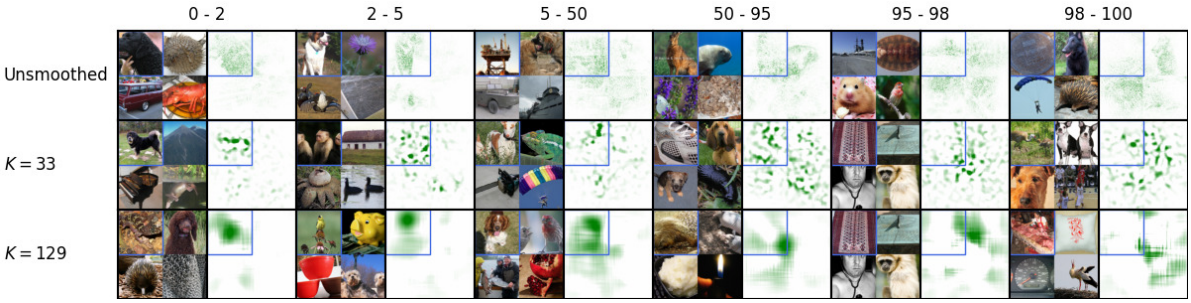


Figure 21. **Examples from each AggAtt bin after smoothing IxG attributions on GridPG using VGG11.** From each bin, the image and its attribution at the median position are shown.

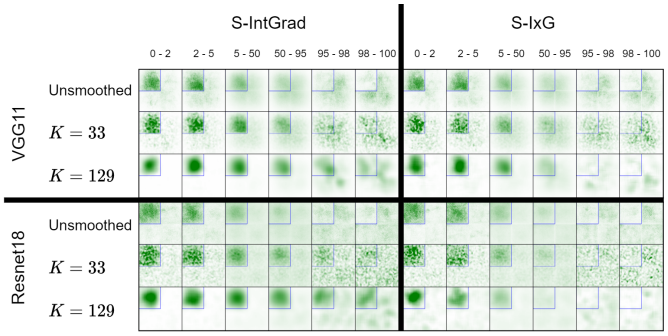
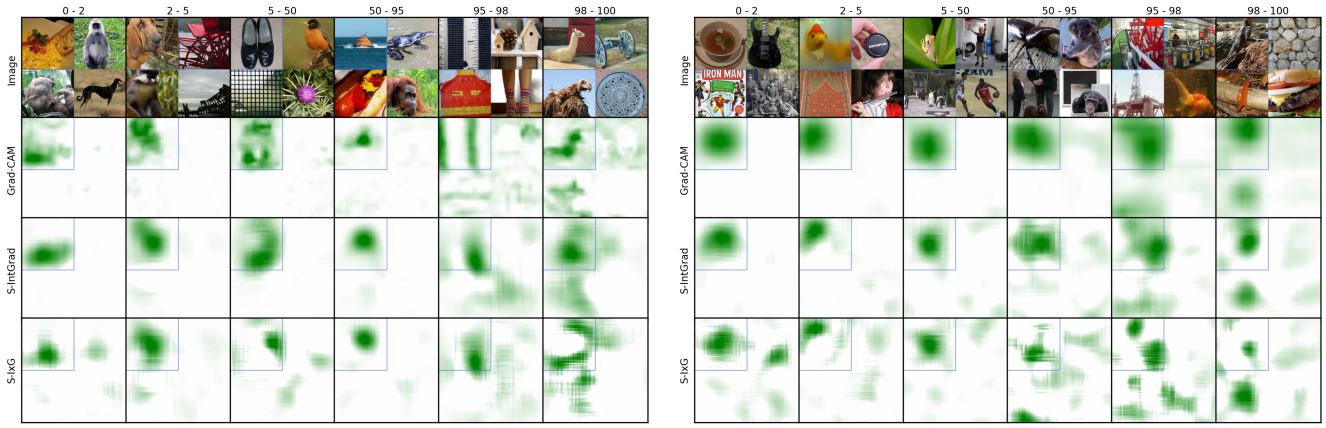


Figure 22. **Impact of smoothing IntGrad and IxG attributions using VGG11 and Resnet18 visualized through the AggAtt evaluation on GridPG.**

D.3. Comparing Grad-CAM with S-IntGrad and S-IxG

We now compare Grad-CAM at the final layer with S-IntGrad and S-IxG at the input layer with $K = 129$ on the same set of examples (Fig. 23). We pick an example from each AggAtt bin of Grad-CAM, and evaluate all three methods on them. From Fig. 23, we observe that the three methods produce visually similar attributions across the AggAtt bins. While the attributions of S-IntGrad and S-IxG are somewhat coarser than Grad-CAM, particularly for the examples in the first few bins, they still concentrate around similar regions in the images. Interestingly, they perform similarly even for examples where Grad-CAM does not localize well, i.e., in the last two bins. Finally, we again see that S-IxG using Resnet18 performs relatively worse as compared to the other methods (as also seen in Tab. 2).



(a) VGG11 (b) Resnet18

Figure 23. **Example attributions from each AggAtt bin of Grad-CAM at the final layer compared with corresponding attributions from S-IntGrad and S-IxG at the input layer with $K = 129$, using Resnet18 on GridPG.** We observe that S-IntGrad and S-IxG show visually similar examples to Grad-CAM across bins for VGG11. While S-IntGrad also performs similarly for Resnet18, S-IxG produces more noisy attributions.

E. Quantitative Evaluation on All Layers

For a fair comparison, we evaluated each method at the input, a middle layer, and the final layer of the network. The middle layer was chosen as a representative to visualize the trends in the localization performance across the network. Figs. 24 and 25 show the results on evaluating at each convolutional layer of VGG11 [S14] and each layer block of Resnet18 [S2]. We find that the performance on the remaining layers is consistent with the trend observed from the three chosen layers in our experiments.

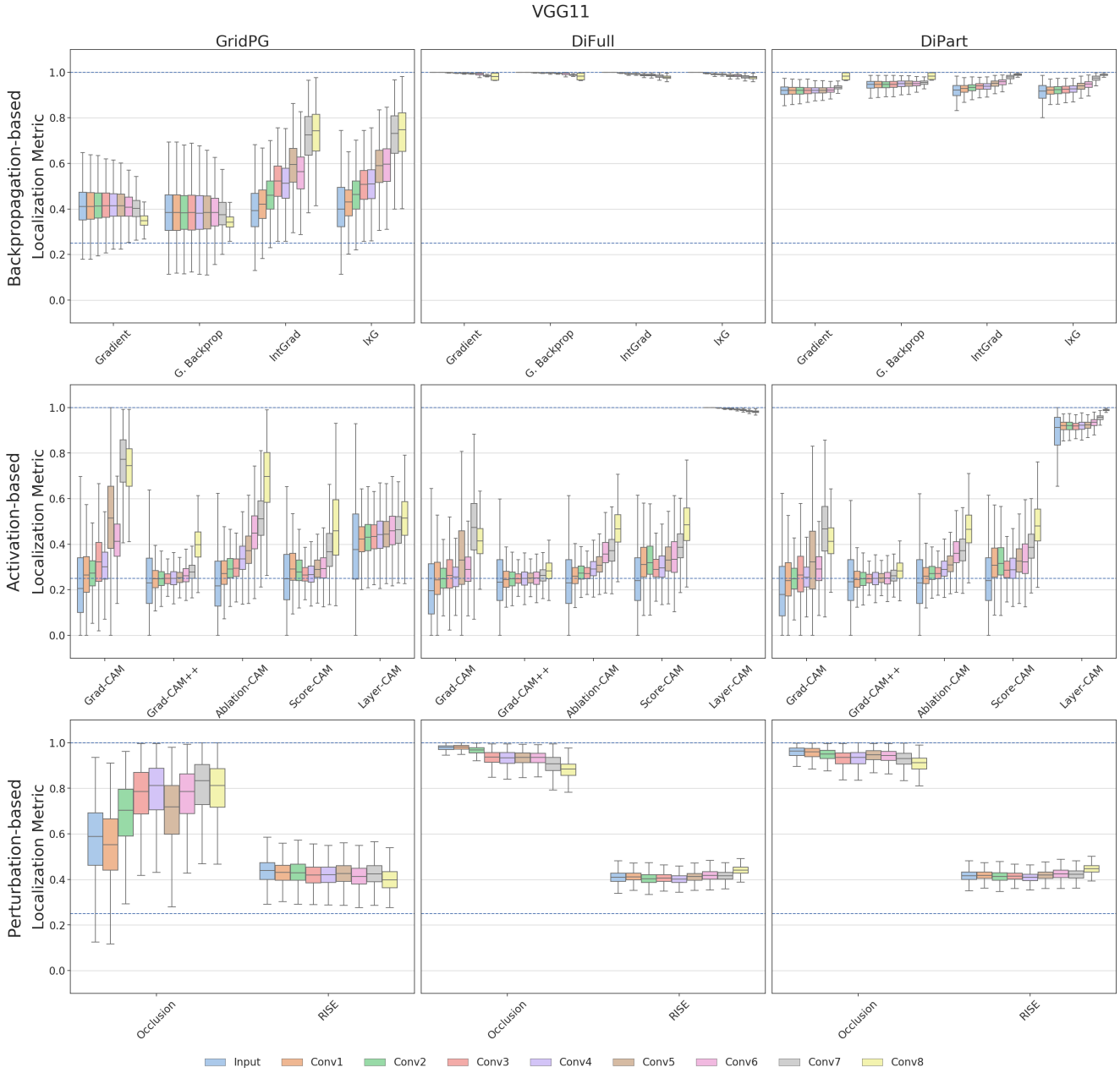


Figure 24. **Quantitative Results for VGG11 across all convolutional layers**. We evaluate the localization scores each attribution method at the input and each convolutional layer of VGG11, on each of GridPG, DiFull, and DiPart. *Top:* Backpropagation-based methods. *Middle:* Activation-based methods. *Bottom:* Perturbation-based methods. The two horizontal dotted lines mark localization scores of 1.0 and 0.25, which correspond to perfect and random localization, respectively. We find that the trends in performance corroborate with those seen across the selected input, middle, and final layers in our experiments.

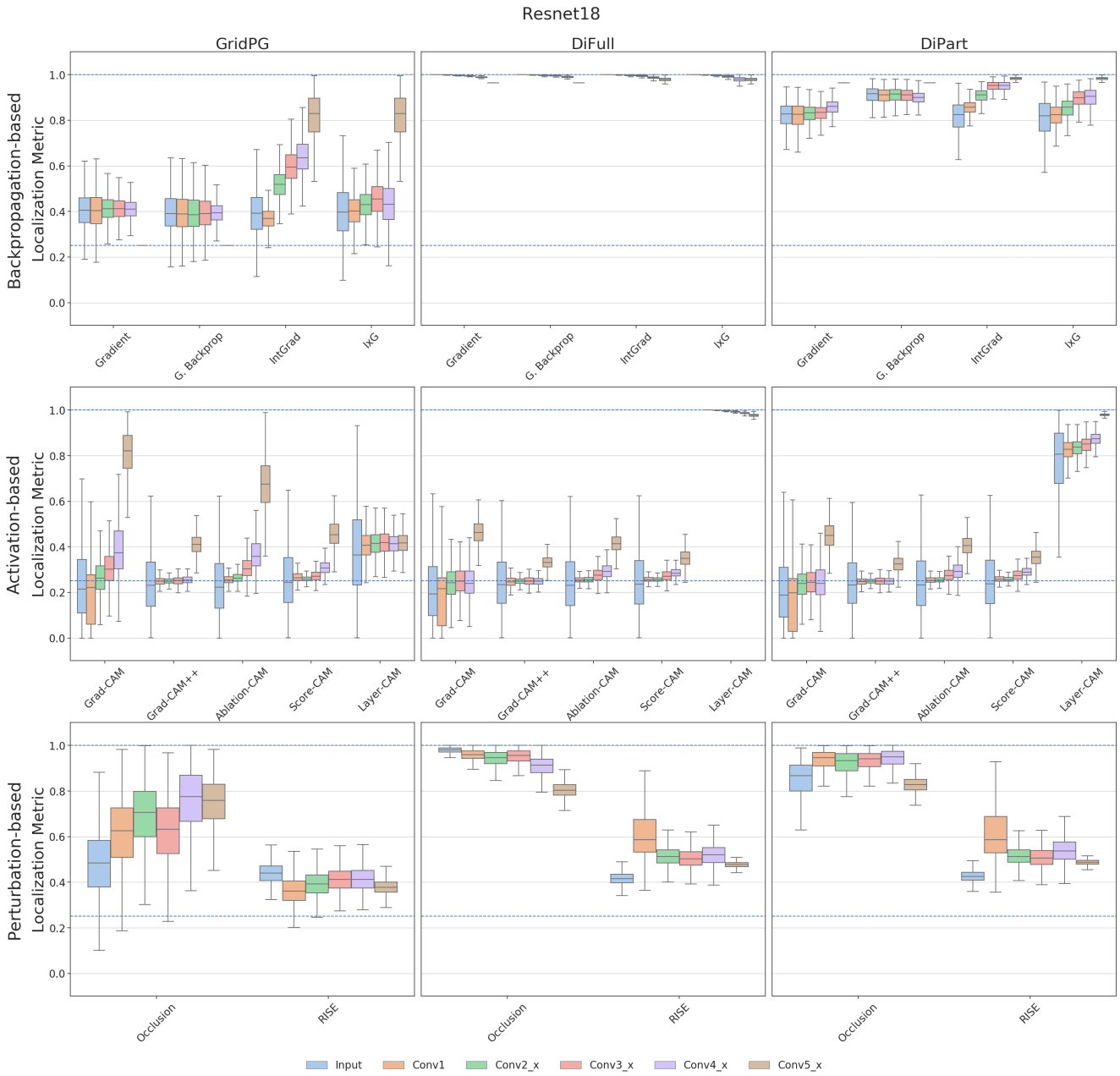


Figure 25. **Quantitative Results for Resnet18 across all convolutional layer blocks**. We evaluate the localization scores each attribution method at the input and each convolutional layer of Resnet18, on each of GridPG, DiFull, and DiPart. *Top:* Backpropagation-based methods. *Middle:* Activation-based methods. *Bottom:* Perturbation-based methods. The two horizontal dotted lines mark localization scores of 1.0 and 0.25, which correspond to perfect and random localization, respectively. We find that the trends in performance corroborate with those seen across the selected input, middle, and final layers in our experiments.

F. Computational Cost

Unlike GridPG [S1], the DiFull setting involves passing each grid cell separately through the network. In this section, we compare the computational costs of GridPG, DiFull, and DiPart, and show that it is similar across the three settings. Let the input be in the form of a $n \times n$ grid. Each setting consists of a CNN module, which obtains features from the input, and a classifier module, which provides logits for each cell in the grid using the obtained features. We analyze each of these modules one by one.

CNN Module: In GridPG and DiPart, the entire grid is passed through the CNN module as a single input. On the other hand, in DiFull, each grid cell is passed separately. This can be alternatively viewed as stacking each of the n^2 grid cells along the batch dimension before passing them through the network. Consequently, the inputs in the DiFull setting have their widths and heights scaled by a factor of $\frac{1}{n}$, and the batch size scaled by a factor of n^2. Since the operations within the CNN module scale linearly with input size, the computational cost for each grid cell in DiFull is $\frac{1}{n^2}$ times the cost for the full grid in GridPG and DiPart. Since there are n^2 such grid cells, the total computational cost for the CNN module of DiFull equals that of GridPG and DiPart.

Classifier Module: The classifier module in the DiFull and DiPart settings consists of n^2 classification heads, each of which receives features corresponding to a single grid cell. On the other hand, the GridPG setting uses a classifier kernel over the composite feature map for the full grid. Let the dimensions of the feature map for a single grid cell be $d \times d$. This implies that in GridPG, using a stride of 1, the classification kernel slides over $((n-1)d+1)^2$ windows of the input, each of which results in a call to the classifier module. In contrast, in DiFull and DiPart, the classifier module is called only n^2 times, one for each head. This shows that the computational cost of DiFull and DiPart for the classifier module and the pipeline as a whole is at most as much as of GridPG.

G. Comparison with SmoothGrad

In our work, we find that smoothing IntGrad [S17] and IxG [S12] attributions with a Gaussian kernel can lead to significantly improved localization, particularly for networks without batch normalization layers [S3]. As discussed in Sec. D.1, we believe this to be because smoothing summarizes the effect of inputs in a local window around each pixel to the output logit, and reduces noisiness of attributions. Prior approaches to address noise in attributions include SmoothGrad [S15], which involves adding Gaussian noise to an input and averaging over attributions from several samples. Here, we compare our smoothing with that of SmoothGrad. Fig. 26 shows that our methods (S-IntGrad, S-IxG) show significantly better GridPG [S1] localization than SmoothGrad on IntGrad and IxG, except in the case of IxG with Resnet18 [S2], where our smoothing does not improve localization likely due to the presence of batch normalization layers. The scores on DiFull decrease to an extent since our Gaussian smoothing allows attributions to "leak" to neighbouring grid cells. These results are corroborated by AggAtt visualizations in Fig. 27. We also note that SmoothGrad requires significantly higher computational cost than our approach, as attributions need to be generated for several noisy samples of each input, and is also sensitive to the choice of hyperparameters such as the noise percentage and the number of samples.

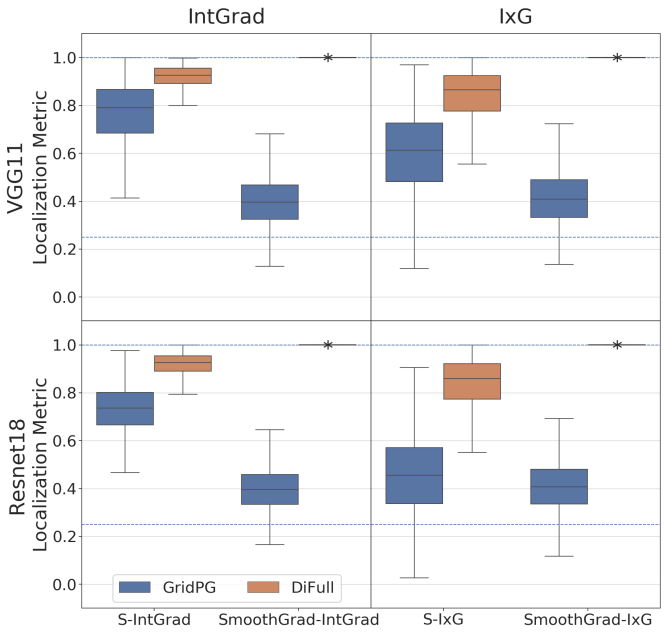


Figure 26. **Quantitative Results comparing our smoothing with SmoothGrad using VGG11**. We evaluate each attribution method at the input layer. For S-IntGrad and S-IxG, we use $K = 129$. For SmoothGrad, we use the best performing configuration after varying the noise percentage from 1% to 30%, and use 15 samples per input. *Top:* Results on VGG11. *Bottom:* Results on Resnet18. We use the "*" symbol to show boxes that collapse to a single point, for better readability.

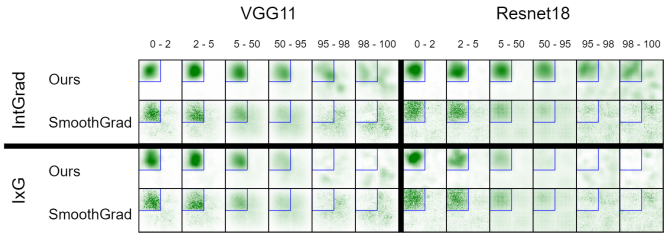


Figure 27. **AggAtt visualizations of our smoothing (S-IntGrad,S-IxG IntGrad) compared with SmoothGrad applied on IntGrad and IxG using VGG11 and Resnet18 on GridPG.**

H. Implementation Details

H.1. Dataset

As described in the paper (Sec. 4), we obtain 2,000 attributions for each attribution method on each of GridPG [S1], DiFull, and DiPart, using inputs consisting of four subimages arranged in 2×2 grids. For GridPG, since we evaluate on all four subimages, we do this by constructing 500 grid images after randomly sampling 2,000 images from the validation set. Each grid image contains subimages from four distinct classes. On the other hand, for DiFull and DiPart, we place images of the same class at the top-left and bottom-right corners to test whether an attribution method simply highlights class-related features, irrespective of them being used by the model. Therefore, we evaluate only on these two grid locations. In order to obtain 2,000 attributions as with GridPG, for these two settings, we construct 1,000 grid images by randomly sampling 4,000 images from the validation set.

H.2. Models and Attribution Methods

We implement our settings using PyTorch [S8], and use pretrained VGG11 [S14] and Resnet18 [S2] models from Torchvision [S8]. We use implementations from the Captum library [S5] for Gradient [S13], Guided Backprop [S16], IntGrad [S17], and IxG [S12], and from [S1] for Occlusion [S18] and RISE [S9]. For Gradient and Guided Backprop, the absolute value of the attributions are used. All attributions across methods are summed along the channel dimensions before evaluation.

Occlusion involves sliding an occlusion kernel of size K with stride s over the image. As the spatial dimensions of the feature maps decreases from the input to the final layer, we select different values of K and s for each layer. In our experiments, we use $K = 16, s = 8$ for the input, and $K = 5, s = 2$ for the middle and final layers.

RISE generates attributions by occluding the image using several randomly generated masks and weighing them based on the change in the output class confidence. In our experiments, we use $M = 1000$ masks. We use fewer masks than [S9] to offset the increased computational cost from using 448×448 images, but found similar results from a subset of experiments with $M = 6000$.

H.3. Localization Metric

In our quantitative evaluation, use the same formulation for the localization score as proposed in GridPG (Sec 3.1.1 in the paper). 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:

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

However, L_i is undefined when the denominator in Eq. (2) is zero, i.e., $\sum_{j=1}^{n^2} \sum_{p \in x_j} A^+(p) = 0$. This can happen, for instance, when all attributions for an input are negative. To handle such cases, we set $L_i = 0$ in our evaluation whenever the denominator is zero.

H.4. AggAtt Visualizations

To generate our AggAtt visualizations, we sort attribution maps in the descending order of the localization score and bin them into percentile ranges to obtain aggregate attribution maps (Sec 3.2 in the paper). However, we observe that when evaluating on DiFull, the backpropagation-based attribution methods show perfect localization (Sec 5.1 in the paper), and all attributions share the same localization score. In this scenario, and in all other instances when two attributions have the same localization score, we break the tie by favouring maps that have stronger attributions in the target grid cell. We do this by ordering attributions with the same localization score in the descending order of the sum of attributions within the target grid cell, i.e., the numerator in Eq. (2).

Further, when producing the aggregate maps, we normalize the aggregate attributions using a common normalizing factor for each method. This is done to accurately reflect the strength of the average attributions across bins for a particular method.

I. Evaluation on CIFAR10

In addition to ImageNet [S10], we also evaluate using our settings on CIFAR10 [S6]. In this section, we present these results, and find similar trends in performance as on ImageNet. We first describe the experimental setup (Sec. I.1) used, and then show the quantitative results on GridPG [S1], DiFull, and DiPart (Sec. I.2) and some qualitative results using AggAtt (Sec. I.3).

I.1. Experimental Setup

Network Architecture: We use a modified version of the VGG11 [S14] architecture, with the last two convolutional layers removed. Since the CIFAR10 inputs have smaller dimensions (32×32) than Imagenet (224×224), using all the convolutional layers results in activations with very small spatial dimensions, which makes it difficult to apply attribution methods at the final layer. After removing the last two convolutional layers, we obtain activations at the new final layer with dimensions 4×4 before pooling. We then perform our evaluation at the input (Inp), middle layer (Conv3) and the final layer (Conv6).
Data: We construct grid datasets consisting of 2×2 and 3×3 grids using images from the validation set classified correctly by the network with a confidence of at least 0.99. We obtain 4,000 (resp. 4,500) attributions for each method from the 2×2 (resp. 3×3) grid datasets respectively. As with ImageNet (Sec. H.1), we evaluate on all grid cells for GridPG and only at the top-left and bottom-right corners on DiFull and DiPart. To obtain an equivalent 4,000 (resp. 4,500) attributions from using just the corners on DiFull and DiPart, we randomly sample 8,000 (resp. 20,250) images for the 2×2 (3×3) grid datasets, and construct 2,000 (2,250) composite images. Note that the CIFAR10 validation set only has a total of 10,000 images. Since we only evaluate at the two corners, we allow subimages at other grid cells to repeat across multiple composite images. However, no two subimages are identical within the same composite image.

I.2. Quantitative Evaluation on GridPG, DiFull, and DiPart

The results of the quantitative evaluation can be found in Fig. 28 for both 2×2 grids (left) and 3×3 grids (right). We observe that all methods perform similarly as on ImageNet (Fig. 9). Since localizing on 3×3 grids poses a more challenging task, we observe generally poorer performance across all methods on that setting.

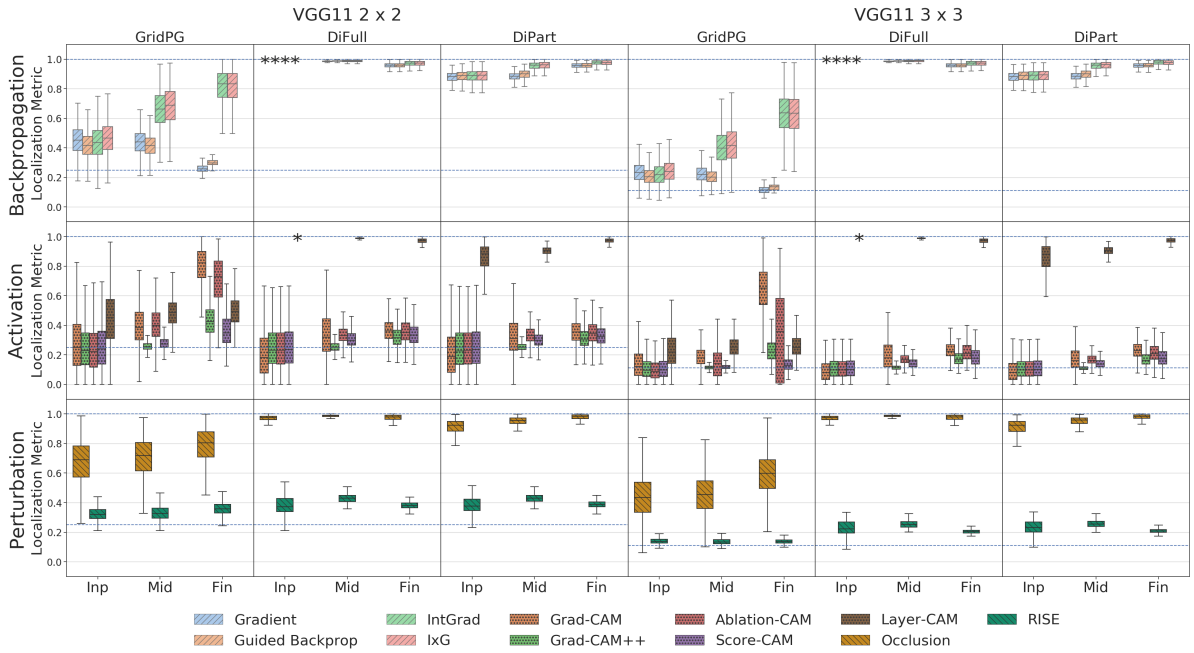


Figure 28. **Quantitative Results on CIFAR10 using VGG11**. We evaluate each attribution method at the input (Inp), middle (Mid), and final (Fin) convolutional layers, on each of GridPG, DiFull, and DiPart using 2×2 (left) and 3×3 (right) grids. *Top:* Results on backpropagation-based methods. *Middle:* Results on activation-based methods. *Bottom:* Results on perturbation-based methods. The two horizontal dotted lines mark localization scores that correspond to perfect and random localization, respectively, which equal scores of 0.25 and 0.11 respectively for 2×2 and 3×3 grids. We use the "*" symbol to show boxes that collapse to a single point, for better readability.

I.3. Qualitative Results using AggAtt

In Fig. 29, we show AggAtt evaluations on 3×3 grids for a method each from the set of backpropagation-based (IxG [S12]), activation-based (Grad-CAM [S11]), and perturbation-based (Occlusion [S18]) methods. Further, we show examples of attributions at the input and final layer on GridPG for these methods (Figs. 30 and 31). We see that these show similar trends in their performance as on ImageNet (Sec. B).

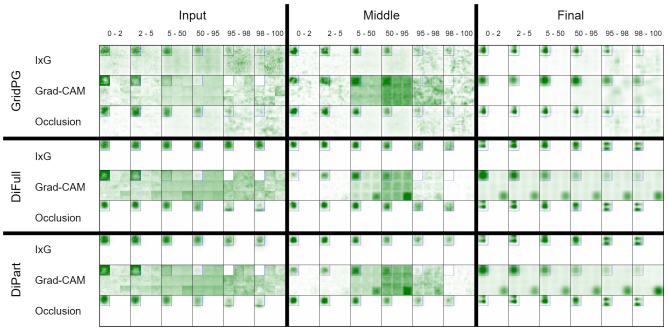


Figure 29. **AggAtt Evaluation on GridPG for all methods at the input, middle, and final layers using VGG11 with the 3×3 grid dataset.**

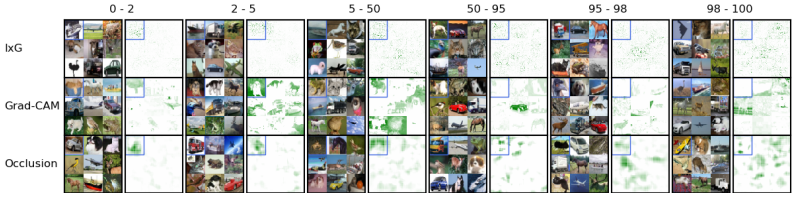


Figure 30. **Examples from each AggAtt bin for each method at the input layer on GridPG using VGG11 on 3×3 grids.** From each bin, the image and its attribution at the median position are shown.

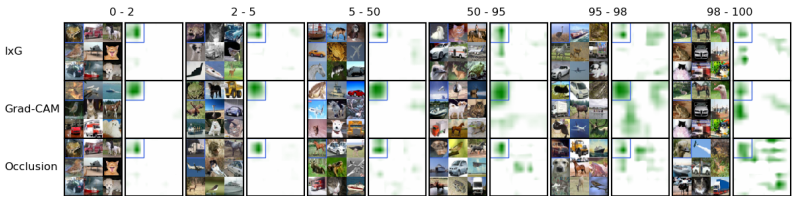


Figure 31. **Examples from each AggAtt bin for each method at the final layer on GridPG using VGG11 on 3×3 grids.** From each bin, the image and its attribution at the median position are shown.

References

[S1] Moritz Böhle, Mario Fritz, and Bernt Schiele. Convolutional Dynamic Alignment Networks for Interpretable Classifications. In *CVPR*, pages 10029–10038, 2021. 2, 3, 9, 15, 16, 17, 18

[S2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, pages 770–778, 2016. 2, 4, 9, 11, 13, 16, 17

[S3] Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *ICML*, pages 448–456, 2015. 16

[S4] 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, 3

[S5] Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, and Orion Reblitz-Richardson. Captum: A unified and generic model interpretability library for PyTorch. *arXiv preprint arXiv:2009.07896*, 2020. 17

[S6] Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images. 2009. 1, 18

[S7] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the Effective Receptive Field in Deep Convolutional Neural Networks. In *NeurIPS*, 2016. 11

[S8] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *NeurIPS*, 2019. 17

[S9] Vitali Petsiuk, Abir Das, and Kate Saenko. RISE: Randomized Input Sampling for Explanation of Black-box Models. In *BMVC*, 2018. 3, 17

[S10] 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. 1, 18

[S11] 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. 1, 3, 9, 11, 19

[S12] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning Important Features Through Propagating Activation Differences. In *ICML*, pages 3145–3153, 2017. 3, 9, 11, 16, 17, 19

[S13] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. In *ICLRW*, 2014. 2, 3, 17

[S14] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *ICLR*, 2015. 2, 3, 9, 11, 13, 17, 18

[S15] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. SmoothGrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017. 1, 16

[S16] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for Simplicity: The All Convolutional Net. In *ICLRW*, 2015. 2, 3, 17

[S17] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic Attribution for Deep Networks. In *ICML*, pages 3319–3328, 2017. 3, 9, 11, 16, 17

[S18] Matthew D Zeiler and Rob Fergus. Visualizing and Understanding Convolutional Networks. In *ECCV*, pages 818–833, 2014. 3, 9, 17, 19