# A Novel Visual Interpretability for Deep Neural Networks by Optimizing Activation Maps with Perturbation

**Qinglong Zhang, Lu Rao, Yubin Yang**
State Key Laboratory for Novel Software Technology at Nanjing University
{wofmanaf, raoluSmile}@smail.nju.edu.cn, yangyubin@nju.edu.cn

## Abstract

Interpretability has been regarded as an essential component for deploying deep neural networks, in which the saliency-based method is one of the most prevailing interpretable approaches since it can generate individually intuitive heatmaps that highlight parts of the input image that are most important to the decision of the deep networks on a particular classification target. However, heatmaps generated by existing methods either contain little information to represent objects (perturbation-based methods) or cannot effectively locate multi-class objects (activation-based approaches). To address this issue, a two-stage framework for visualizing the interpretability of deep neural networks, called Activation Optimized with Perturbation (AOP), is designed to optimize activation maps generated by general activation-based methods with the help of perturbation-based methods. Finally, in order to obtain better explanations for different types of images, we further present an instance of the AOP framework, Smooth Integrated Gradient-based Class Activation Map (SIGCAM), which proposes a weighted GradCAM by applying the feature map as weight coefficients and employs I-GOS to optimize the base-mask generated by weighted GradCAM. Experimental results on common-used benchmarks, including deletion and insertion tests on ImageNet-1k, and pointing game tests on COCO2017, show that the proposed AOP and SIGCAM outperform the current state-of-the-art methods significantly by generating higher quality image-based saliency maps.

## Introduction

Understanding and explaining deep learning methods have been attracting increasing attention in research communities since it helps to construct the trust of black-box models when making crucial decisions, particularly in applications including quantitative transaction, autopilot, and medical image analysis, etc.

One critical issue of understanding deep neural network is to explicitly generate images that can shed light on parts which are most related to the deep neural networks' decision. A common way is to back-propagate the gradient or its variants to the input image to decide which pixels or regions are more relevant to the change of the prediction (Qi, Khorram, and Li 2020; Sundararajan, Taly, and Yan 2017; Smilkov et al. 2017). However, these changes do not necessarily indicate that the pixels or regions really have a significant impact on the prediction (An example is illustrated in Figure 1).

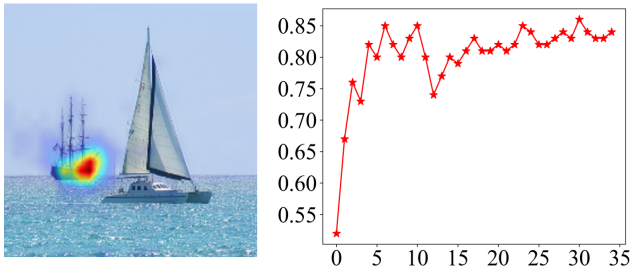


Figure 1: classification score (after softmax) on Densnet201: "catamaran": 0.521; "pirate": 0.068. We gradually remove "pirate" according to its mask(Left) and test its effect on "catamaran" (Right). The deletion test shows that the removal of "pirate" significantly affects the classification score of "catamaran", although it is irrelevant to "catamaran".

Other approaches, such as activation-based methods including GradCAM (Selvaraju et al. 2017) and GradCAM++ (Chattopadhyay et al. 2018), use back-propagation gradients as weights to combine the forward feature maps, which favor distinguishing objects from the background, and thus provide explanations with fine-grained details. However, these methods may capture too much meaningless information since the feature maps are not necessarily related to the target category. On the other hand, gradients are class-discriminative, which means optimizing gradients can better identify different object categories. Perturbation-based methods such as the method in (Fong and Vedaldi 2017)(to simplify, we call it EMP) and I-GOS (Qi, Khorram, and Li 2020) adopt gradients as masks and use them to edit an image to convert gradient optimization into a meta-predictor problem. Nevertheless, heatmaps generated by perturbation-based approaches may contain very little information used to represent an object. It has been widely demonstrated that properly cascaded different kinds of deep learning methods can deliver continuous optimizing results (Yan et al.

2017; Marcetic, Soldic, and Ribaric 2017). Therefore, to gain highly network decision-related explanations with adequate information, a general two-stage framework, Activation Optimized with Perturbation (AOP) is proposed, by utilizing perturbation-based approaches to optimize the base-bask generated by activation-based methods.

Although GradCAM (Selvaraju et al. 2017) and GradCAM++ (Chattopadhyay et al. 2018) get well performance on "single-objective" images, they still have limitations. For example, GradCAM's performance drops when localizing "multi-objective, single-class" images (Chattopadhyay et al. 2018), while GradCAM++ may capture unrelated information on "multi-objective, multi-class" images. Examples are illustrated in Figure 3. In order to solve these problems, based on the AOP framework, this paper introduces Smooth Integrated Gradient-based Class Activation Map (SIGCAM), which firstly proposed a weighted GradCAM (WGradCAM) to generate well performed base-mask on different types of images, and then employs I-GOS (Qi, Khorram, and Li 2020) to optimize the generated base-mask. In addition, to reduce computational overhead and make the optimizing process converge faster, a startup strategy is adopted, and the pixels with less importance are filtered.

The key contributions in this paper are summarized as follows: 1) we introduce a highly flexible two-stage framework AOP to combine the advantages of activation-based methods to capture adequate information and the ability of perturbation-based methods to generate network decision-related masks. 2) we propose Weighted GradCAM, an extension of GradCAM, by applying the feature map as weight coefficients, which is similar to GradCAM++ but simpler and more effective. 3) we present SIGCAM, a special instance of AOP, which employs I-GOS to optimize the proposed Weighted GradCAM. 4) experimental results in insertion and deletion metrics (Petsiuk, Das, and Saenko 2018) on ImageNet-1k and pointing game (Zhang et al. 2018) on COCO2017 show that AOP and SIGCAM can greatly surpasses the current SOTA explanation approaches.

## Related Work

In this section, we present a survey of related works that generate visual, image-like explanations. To simplify, we classify them into four categories: 1) Gradient-based Methods; 2) Activation-based Methods; 3) Region-based Methods; 4) Perturbation-based Methods.

**Gradient-based Methods.** These approaches calculate attributions by back-propagating the predicted score in each layer of the network, and then return them to the input features. They usually generate pixel-level attributions and have been widely studied. SmoothGrad (Smilkov et al. 2017) and VarGrad (Adebayo et al. 2018) sought to alleviate noise and visual diffusion for saliency maps by averaging over explanations of noise copies of an input, which visually sharpened explanations. Integrated gradients (Sundararajan, Taly, and Yan 2017) attempted to address "gradient saturation" by estimating the global importance of each pixel, rather than local sensitivity. Guided Backpropagation (Springenberg et al. 2015) set negative gradient zero through a ReLU unit in the back-propagation. Other gradient-based approaches such as DeepLift (Shrikumar, Greenside, and Kundaje 2017) or Excitation BackProp (Zhang et al. 2018) utilized top-down relevancy propagations. Gradient-based are in general faster but tend to achieve lower quality saliency maps.

**Activation-based Methods.** These methods combine activations from convolutional layers (which are usually called "target layers") to form an explanation. CAM (Zhou et al. 2016) and GradCAM (Selvaraju et al. 2017) used a linear combination of activations to form a heatmap with fine-grained details of the predicted class. GradCAM++ (Chattopadhyay et al. 2018) adopted a weighted combination of the positive partial derivatives of the target layers' feature maps with respect to a specific class score as weights to generate a visual explanation for the corresponding class label. To better visualize the effect of activation-based approaches, the generated heatmaps should be unsampled to the same resolution of the original input image.

**Region-based methods.** These approaches estimate the feature importance of different regions. RISE (Petsiuk, Das, and Saenko 2018) took a linear combination of random masks where the weights were computed from the score of the target class corresponding to the respective masked inputs. Dasom Seo et al (Seo, Oh, and Oh 2020) obtained a more class-discriminative and visually pleasing map than RISE by fusing saliency maps generated from multi-scale segmentations. Similar to (Seo, Oh, and Oh 2020), XRAI (Kapishnikov et al. 2019) firstly over-segmented the image, then iteratively test the importance of each region, coalescing smaller regions into larger segments based on attribution scores. The main difference between XRAI and (Seo, Oh, and Oh 2020) is that XRAI uses Integrated Gradients to generate attributions. Region-based approaches usually generate better human-interpretable visualizations but are less efficient than gradient-based approaches and activation-based methods.

**Perturbation-based Methods.** Such approaches directly estimate the impact of a feature subset on the output. LIME (Ribeiro, Singh, and Guestrin 2016) explained the predictions of any classifier in an interpretable and reliable manner by learning an interpretable model locally around the predictions. EMP (Fong and Vedaldi 2017) directly edited the image to learn the location at which a model focused on by discovering the parts of an image that most affect its output score when it is perturbed, making it interpretable and testable. I-GOS (Qi, Khorram, and Li 2020) extended EMP (Fong and Vedaldi 2017) by computing descent directions based on the integrated gradients to avoid local optima, and achieved a speed-up of convergence. Since perturbation-based methods require multiple queries to the model, they are usually slower than other methods except for the region-based ones.

## The Proposed Methods

In this section, we first introduce a two-stage saliency map generating framework AOP, which achieves continuous optimization of prediction results by cascading different kinds of saliency methods. Then, we propose WGradCAM, which extends GradCAM by applying the feature map as weight coefficients. And Finally, we present SIGCAM, a particular

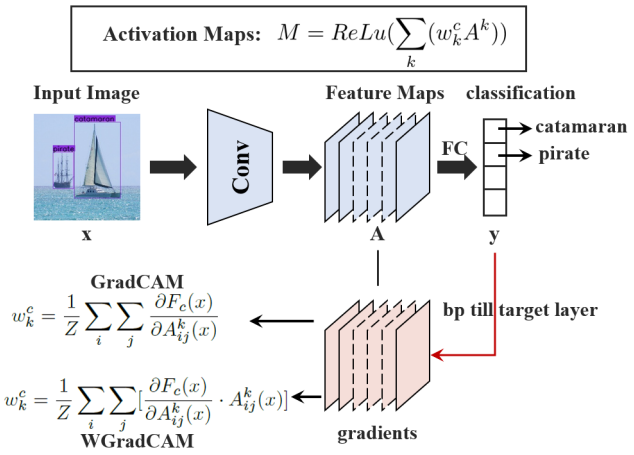


Figure 2: An overview of GradCAM and WGradCAM with their respective computation expressions.

instance of AOP, which uses WGradCAM to generate the basic saliency maps and optimizes them using I-GOS.

## Optimizing Activation with Perturbation

The principle behind AOP is straightforward. Perturbation-based methods allow us to directly estimate the impact of pixels of the output and generate more human-friendly saliency maps. However, they are less efficient, and the optimizing process needs hundreds of iterations. Meanwhile, activation-based methods can quickly generate saliency maps, which capture more information about the target object. Therefore, the motivation of AOP is applying activation-based methods to generate a base-mask and utilizing a perturbation-based method to optimize it. Note that the base-mask of AOP can be generated by different kinds of activation-based methods, and the optimizing process can be conducted by various perturbation-based approaches, making the AOP framework highly extensible.

## Weighted GradCAM

While Grad CAM++ generates a broader salient region than GradCAM and is more suitable for the base-mask, it achieves a worse deletion and insertion score. On the other hand, GradCAM++ adopts higher-order derivatives to take a weighted average of the pixel-wise gradients, which is time-consuming. To address this issue, we modify the structure of GradCAM and make the generated heatmaps more discriminative. Different from Grad CAM++, the proposed method WGradCAM adopts a more straightforward but more effective way, by applying the feature map as weight coefficients.

**Structure of WGradCAM.** The main difference between WGradCAM and GradCAM is the calculation of gradient weights $w_k^c$ for a particular class $c$ and activation map $A_k$ (shown in Figure 2).

Specifically, GradCAM defines the neuron importance weights $w_k^c$ as

$$w_k^c = \frac{1}{Z}\sum_i\sum_j \frac{\partial F_c(x)}{\partial A_{ij}^k(x)} \tag{1}$$

where $Z$ is the number of pixels in the activation map.

However, WGradCAM applies the feature map in Eq. 1 in a pixel-wise way, that is,

$$w_k^c = \frac{1}{Z}\sum_i\sum_j [\frac{\partial F_c(x)}{\partial A_{ij}^k(x)} \cdot A_{ij}^k(x)] \tag{2}$$

Similar to GradCAM, WGradCAM performs a weighted combination of forward activation maps $A^k$, and calculate a mask $M$ by ReLU.

$$M = ReLu(\sum_k (w_k^c A^k)) \tag{3}$$

We scale $M$ into $[0.0, 1.0]$ by utilizing Min-Max normalization,

$$M = \frac{M - min(M)}{max(M) - min(M)} \tag{4}$$

Then, $M$ can be optimized using perturbation-based approaches.

## SIGCAM

Directly optimizing $M$ generated by WGradCAM is time-consuming since it contains too many non-zero pixels. Besides, too many pixels in the base-mask get high values, which are not human-friendly. To deal with this problem, we use $filter(M < \theta)$ to filter the pixels less than the threshold $\theta$ (setting these pixels 0) and use a modulating factor parameter $\lambda$ to control the initial value to a specific range as follows.

$$M = \lambda(filter(M < \theta)) \tag{5}$$

Then, $M$ can be upsampled to a matrix whose size is smaller than the shape of the input image (e.g., the shape of $M$ is $7 \times 7$ when using ResNet50 as the classification model, it can be upsampled to $28 \times 28$ and then used as the base-mask). Since this paper considers the process of fine-tuning $M$ as a "deletion game", which aims at finding the smallest mask $M^*$ that causes the score $F_c(\Phi(I_0, \text{up}(M, (H, W)))) \leq F_c(I_0)$ to drop significantly, the base-mask should be $M = 1 - \text{up}(M, (h_1, w_1))$, where $\text{up}(M, (h_1, w_1))$ means upsampling $M$ to size $(h_1, w_1)$. $\Phi(I_0, up(M, (H, W))) = I_0 \odot \text{up}(M, (H, W)) + \tilde{I}_0 \odot (1 - \text{up}(M, (H, W)))$ and $\tilde{I}_0$ is a baseline image with the same shape as $I_0$, which should have a low score on the class $c$.

Finally $M^*$ can be formulated to minimize the following objective function:

$$\begin{aligned} M^* = &\text{argmin } F_c(\Phi(I_0, \text{up}(M, (H, W)))) \\ &+ \lambda_1 \|1 - M\| + \lambda_2 \text{TV}(M) \end{aligned} \tag{6}$$

where $\|1 - M\|$ is the $\ell_1$ regularization which encourages most of the mask to be turned off and $\text{TV}(M)$ is a second term total-variation norm to make $M$ more piece-wise smooth. $\lambda_1$ and $\lambda_2$ are hyperparameter.

Similar to I-GOS, the proposed SIGCAM employs a line-search based gradient-project method to make each computation of the integrated gradients maximally.

In order to make the optimizing process smoother, a startup strategy similar to warmup (Goyal et al. 2017) is designed. Specifically, in the first few iterations, smaller Gaussian blur parameters are used to obtain $\tilde{I}_0$, then larger ones are applied to optimize $M$. Since we only need to fine-tune $M$, Eq. 6 will converge much faster than I-GOS.

# Experiments

## Experimental Setup

**Datasets.** We use the validation split of ImageNet-1k (which contains 50, 000 images) and the val2017 split of COCO2017 (containing 5, 000 images) to test the performance of AOP and SIGCAM. COCO2017 contains 80 object categories, where many images are "multi-objective, multi-class", making it challenging for pointing games. In order to make the test results more meaningful, we use the provided segmentation masks instead of commonly used bounding boxes for COCO2017 as the ground truth. It is observed that bounding boxes are not suitable for pointing game since a pixel located in the bounding boxes does not necessarily mean that it has some relationship with the object. For ImageNet-1k, we use a set of pre-trained torchvision models[1]: Densenet201, Inception_v3, and Vgg19. For COCO2017, we follow the necessary training procedure for image classification to fine-tuning the output layer of a pre-trained ResNet50 on train2017 split and validate it on val2017 split. Specifically, only the output layer is fine-tuned using BCEWithLogitsLoss. The training converges after dozens of iterations, and the final validation performance (in %) for classification is Loss:9.07 and Precision:86.53. For both datasets, all images are resized to $224 \times 224$.

**Baselines.** We compare SIGCAM with the current SOTA methods, including activation-based methods: GradCAM (Selvaraju et al. 2017), GradCAM++ (Chattopadhyay et al. 2018), region-based method: XRAI (Kapishnikov et al. 2019) and perturbation-based method: I-GOS (Qi, Khorram, and Li 2020). For GradCAM, GradCAM++ and WGradCAM, the target layer selected in this paper is the layer closest to the linear or classifier module(i.e., Resnet50: 'layer4.2', Densenet201: 'features.norm5', Inception_v3: 'Mixed_7c', and Vgg19: 'features.36'). For XRAI, the segmentation method used in this paper is SLIC in the skimage python package[2] instead of Felzenszwalb, because we reported better results using SLIC on XRAI. The scale segment parameter is set in range $[50, 100, 150, 250, 500]$. We follow the original setup of I-GOS. Threshold $\theta$ and modulating factor $\lambda$ in Eq. 5 on ImageNet-1k is $[0.7, 0.15]$ and on COCO2017 is $[0.6, 0.25]$. The "warmup" iterations to optimize the base-mask is 5, and the size of base-mask is $28 \times 28$. Other parameters are the same as that of I-GOS. For a fair comparison, heatmaps of GradCAM and GradCAM++ will be upsampled to $224 \times 224$.

**Evaluation metrics.** We follow (Petsiuk, Das, and Saenko 2018) to conduct deletion and insertion tests to evaluate saliency maps generated by different approaches. The intuition behind the deletion metric is that the removal of pixels/regions most relevant to a class will cause the classification score to drop significantly. Insertion metric, on the other hand, starts with a blurred image and gradually re-introduces content, which produces more realistic images and has the additional advantage of mitigating the impact of adversarial attack examples. In detail, for the deletion metric, we gradually replace $N$ pixels in the original image with a highly blurred version of the original image each time according to the values of the saliency map until no pixels left. Contrary to the deletion metric, the insertion metric replaces $N$ pixels of the blurred image with the original one until the image is well recovered. We calculate the area under curve (AUC) of the classification score after Softmax as a quantitative indicator. Blur method used in this paper is Gaussian Blur with $kernel\_size = 51$ and $sigma = 50$. Besides, we provide the $over - all$ score to comprehensively evaluate the deletion and insertion results, which can calculate by $AUC(insertion) - AUC(deletion)$ .

Pointing game measures the localization accuracy $Acc = \frac{\#Hits}{\#Hits + \#Misses}$ for each object category (if the most salient pixel lies inside the annotated segmentation mask of an object, it is counted as a hit). The overall performance is measured by the mean accuracy across different categories. Specifically, we only test classes with classification score $F_c(I_0) > 0.5$ in each image since many of the images of COCO2017 contain multiple object categories, making it time-consuming to test all classes. Besides, $F_c(I_0) < 0.5$ provided lower confidence to judge an object, which may make the results less reliable.

## Results and Discussion

**Ablation Studies.** We conduct ablation studies to estimate the performance of AOP. To make the experimental results more reliable, we apply 3 combinations, that is, GradCAM (as a base-mask) with I-GOS, GradCAM with EMP (Fong and Vedaldi 2017) and SIGCAM (WGradCAM with I-GOS). We randomly sample 5,000 images from the ImageNet-1k validation dataset and use Inception_v3 as the base classification network. The evaluation metrics used here are deletion and insertion. Results are shown in Table 1.

From Table 1, we can see, compared with GradCAM, WGradCAM achieves 0.68% gains in terms of deletion AUC and 0.79% advantages over insertion AUC. The over-all improvement is 1.48%.

Furthermore, overall scores suggest that AOP can considerably beat each separate method. Specifically, GradCAM and EMP (Fong and Vedaldi 2017) combination gains 2.46% improvement over GradCAM and 14.17% over EMP in terms of over-all AUC. When considering GradCAM and I-GOS combination, the overall improvement will be 8.38% over GradCAM and 8.01% over I-GOS. Furthermore, SIGCAM achieves the best results over other AOP combinations. Compared with the separate implementations, SIGCAM obtains 8.06% improvement over WGradCAM and 9.17% gains over I-GOS in terms of over-all AUC. In conclusion, AOP can achieve continuous better performance by cascading different saliency methods.

[1]https://github.com/pytorch/vision/tree/master/torchvision/models

[2]https://github.com/scikit-image/scikit-image

| AUC | deletion(%) | insertion(%) | **over-all**(%) |
|---|---|---|---|
| **One-Stage Methods** | | | |
| GradCAM | 17.92 | 66.04 | 48.12 |
| **WGradCAM (Ours)** | **17.23** | **66.83** | **49.60** |
| EMP | 12.70 | 49.21 | 36.51 |
| I-GOS | **8.76** | 57.25 | 48.49 |
| **AOP Methods (Ours)** | | | |
| GradCAM + EMP | 11.90 | 62.58 | 50.68 |
| GradCAM + I-GOS | **8.97** | 65.47 | 56.50 |
| SIGCAM | 9.91 | **67.57** | **57.66** |

Table 1: Ablation studies on Inception_v3. Evaluation indicators contain deletion score (smaller AUC is better), insertion score (higher AUC is better), and the comprehensive evaluation results(over-all score, higher is better) on the ImageNet-1k dataset. The best records are marked in bold.

**Validate the Effectiveness of WGradCAM.** In this part, we first visualize the saliency maps of samples from ImageNet-1k to validate the reliability of WGradCAM, then we analyze the intuition behind the multiplying gradients with activations.

As illustrated in Figure 3, WGradCAM generates class-discriminative heatmaps, and can perform well on different types of images, while GradCAM's performance drops when localizing objects on "multi-objective, single-class" images (2nd and 6th images, line 3), and GradCAM++ may capture unrelated information on "multi-objective, multi-class" images (4th image, line 2).

Analysis: In an activation-based method, gradient weights are expected to be high for feature map pixels that contribute to the presence of the object and lower for irrelevant pixels. However, gradients for a CNN can be noisy and tend to vanish due to the saturation problem for Sigmoid function or the zero-gradient region of ReLU function. This can lead to gradients w.r.t input or the internal layer activation looks noisy visually. Therefore, WGradCAM multiplies gradients with feature maps as weights, which can make the output saliency map more dependent on the original feature maps.

**Compared with Different Saliency Methods.** Table 2 shows the performance of all baselines using deletion and insertion tests (on all the 50,000 images).

From Table 2 we can see, I-GOS and SIGCAM unsurprisingly get a better deletion score on all three models, since the optimizing process of I-GOS and SIGCAM is using a deletion game to find the smallest deletion mask that causes the score $F_c(\Phi(I_0, \text{up}(M, (H, W)))) \leq F_c(I_0)$ to drop significantly. GradCAM and GradCAM++ get better insertion scores on Densenet201 and Vgg19, and SIGCAM comes close. It can be explained that the heatmaps of GradCAM and GradCAM++ are larger than other methods. Note that the heatmaps of XRAI are also large enough, but contain more irrelevant image information, making the scores lower; this can be demonstrated from the deletion game. Besides, SIGCAM gets the highest insertion score on Inception_v3. In conclusion, SIGCAM can not only capture the most critical subfeatures but also express adequate information in a smaller region. By the way, all baselines' performance over Vgg19 has a large gap than that over Densenet201 and Inception_v3, this may be due to the classification result of Vgg19 (top-1 error: 27.62) is the worst, while Densent201 and Inception_v3 get better classification results (top-1 error: Densenet201: 22.80, Inception_v3:22.55).

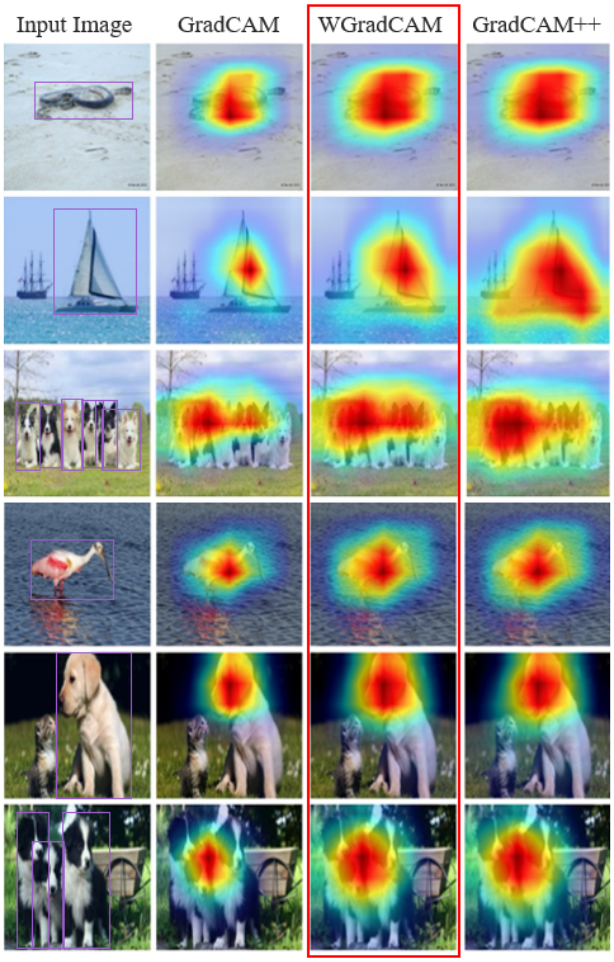


Figure 3: Sample visual explanations on ImageNet-1k generated by GradCAM, WGradCAM(Eq. 3), and GradCAM++ (classification model used is densenet201).

To further explore the details of the insertion and deletion tests, we draw the pixels remove/insert process over the first 5,000 samples of the ImageNet-1k validation dataset. The results are shown in Figure 4.

As shown in Figure 4, SIGCAM outperforms existing related saliency methods, comprehensively considers the results of insertion and deletion. Specifically, I-GOS and SIGCAM perform far better in the deletion test (results are shown in the first line's three images, a smaller AUC means

| Methods | densenet201 | | inception_v3 | | vgg19 | | **over-all** |
|---|---|---|---|---|---|---|---|
| AUC | deletion(%) | insertion(%) | deletion(%) | insertion(%) | deletion(%) | insertion(%) | **over-all(%)** |
| GradCAM | 16.72 | **58.65** | 16.95 | 56.14 | 13.33 | **53.50** | 40.43 |
| GradCAM++ | 17.35 | 57.55 | 16.24 | 53.83 | 14.81 | 50.04 | 37.68 |
| I-GOS | **10.63** | 44.67 | **9.32** | 49.28 | **7.38** | 38.47 | 35.03 |
| XRAI | 24.97 | 45.22 | 16.66 | 54.98 | 15.78 | 43.08 | 28.62 |
| **SIGCAM(Ours)** | **10.97** | 54.61 | **10.22** | **57.94** | **7.80** | 42.54 | **42.04** |

Table 2: Comparative evaluation in terms of deletion (smaller AUC is better) and insertion (higher AUC is better) score on the ImageNet-1k dataset. The over-all score (higher is better) shows that SIGCAM outperform other related methods significantly.

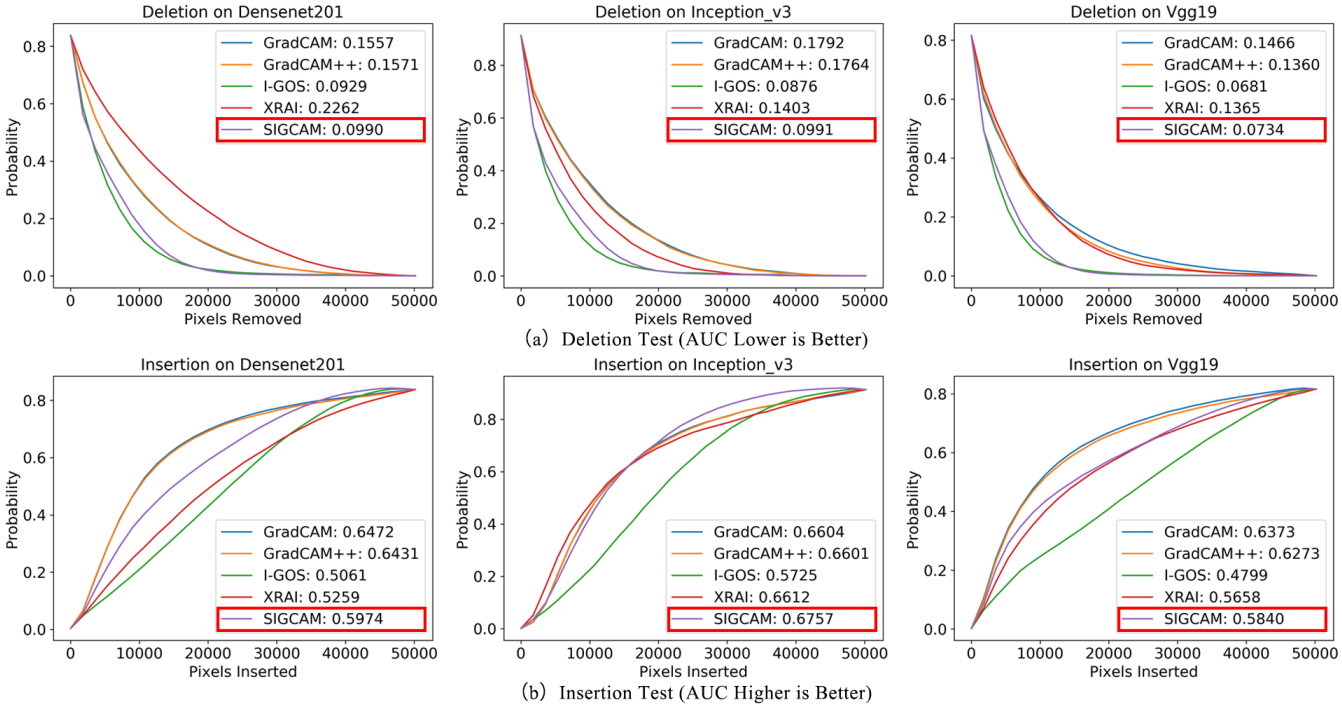


Figure 4: Details of the Insertion & Deletion process. Results of our proposed method SIGCAM is annotated in red rectangles.

that the deleted pixels are more closely related to the network's prediction, i.e., removing these pixels making the prediction score drop significantly). However, I-GOS performs worst over other methods on the insertion test (larger AUC means the prediction probability is higher when inserting these pixels). One possible reason is that saliency maps of I-GOS contain too few pixels. Although removing these pixels can significantly drop the classification score, they do not contain enough information for a classifier to judge a class. Correspondingly, SIGCAM can combine the advantage of I-GOS, which can distinguish the most critical pixels/regions without which the classifier may misjudge a class, and the ability of activation-based methods, which can capture enough information to recover an image.

**Pointing Game on COCO2017.** The results of pointing game on all baselines are shown in Table 3.

From Table 3 we can see, SIGCAM obtains the highest score of all baselines, while I-GOS comes close. Specifically, SIGCAM outperforms other methods with an improvement of $> 2.92\%$ in terms of the mean accuracy.

**Visualization and Interpretation.** Finally, To better interpret the effectiveness of SIGCAM, various approaches including gradient-based methods (Guided-BP (Springenberg et al. 2015), vanilla Back-propagation, Input $\odot$ Gradients (Shrikumar et al. 2016), Integrated Gradients (Sundararajan, Taly, and Yan 2017) and SmoothGrad (Smilkov et al. 2017)), activation-based methods (GradCAM (Selvaraju et al. 2017) and GradCAM++ (Chattopadhyay et al. 2018)), region-based methods (XRAI (Kapishnikov et al. 2019)), and Perturbation-based methods (I-GOS (Qi, Khor-

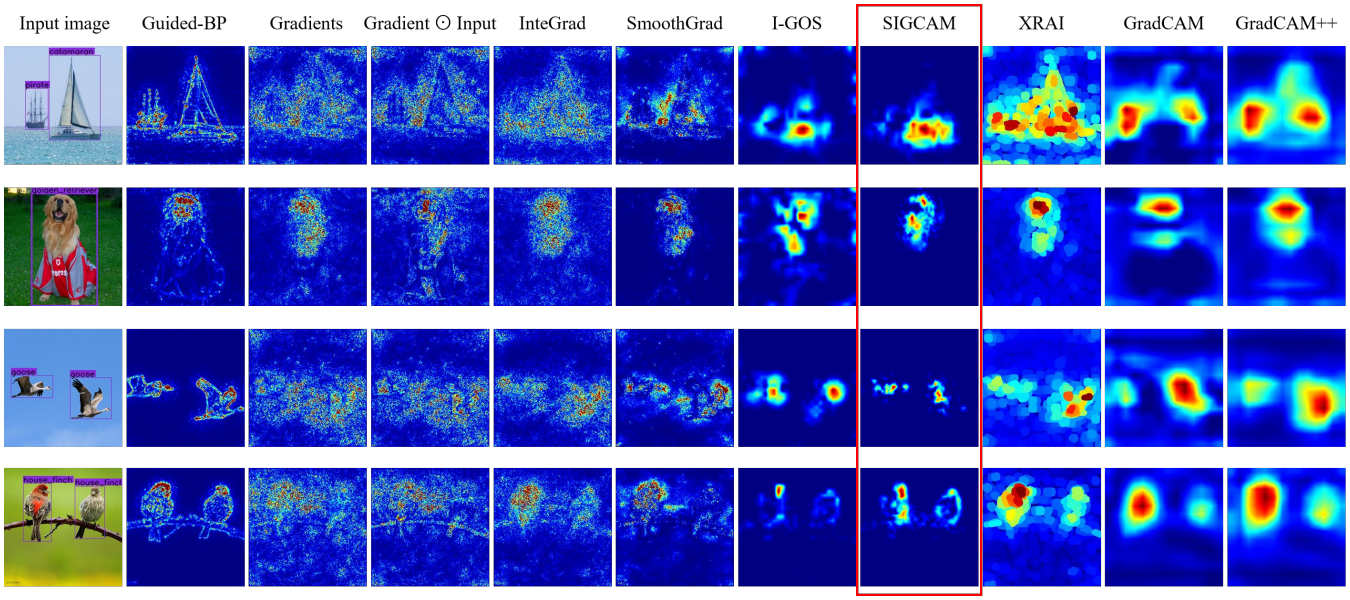


Figure 5: Visualizations comparing SIGCAM vs. various saliency methods on Densenet201. The first columns are Input images with all annotations, others are their heatmaps.

| Methods | Mean Accuracy(%) |
|---|---|
| GradCAM | 25.78 |
| GradCAM++ | 29.91 |
| I-GOS | 35.72 |
| XRAI | 28.12 |
| **SIGCAM (Ours)** | **38.64** |

Table 3: Pointing Game on COCO val2017 split with $F_c(I_0) > 0.5$. Results show that SIGCAM performs consistently better than others with significant improvement.

ram, and Li 2020) and our proposed SigCAM) are applied to visualize the heatmaps of different types of images, i.e., "single-objective", "multi-objective, single-class", and "multi-objective, multi-class". The results are shown in Figure 5.

In order to quantify the evaluation by individual inspection, we define perfect heatmaps should have the following characters: 1) Heatmaps with less noise are better; 2) Heatmaps should best match the target object; 3) Heatmaps should contain adequate information to identify objects.

As is illustrated in Figure 5, heatmaps generated by Guided-BP make more sense to human vision than that generated by other gradient-based methods, such as vanilla Back-propagation, Integrated Gradients, and SmoothGrad, etc. However, too much noise makes Guided-BP less visually friendly than other types of methods. I-GOS's heatmaps contain less noise. However, pixels of these heatmaps have a higher probability of falling outside the segmentation mask, although they tend to be located in the bounding boxes. Heatmaps generated by XRAI are excessively affected by the segmentation result. GradCAM and GradCAM++ generate larger regions, these regions are not that human-friendly, but can capture enough information. SIGCAM generates most human appealing heatmaps above all methods according to 10 humans' evaluation. The heatmaps contain less noise than the one generated by IGOS and capture adequate information to identify the object target like GradCAM and GradCAM++. More importantly, pixels of the SIGCAM's heatmaps are concentrated in the segmentation task, making the heatmaps more human friendly.

In conclusion, SIGCAM can outperform related saliency methods and have great potential in the field of computer visual interpretability.

## Conclusion

In this paper, we propose a two-stage framework AOP, which combines the power of activation-based methods and perturbation-based methods, i.e., optimizing the base-mask generated by activation-based methods to drop unrelated pixels/regions by perturbation-based methods. Then a special instance of AOP, calling SIGCAM is presented to better interpret different types of images. Results of deletion & insertion test and pointing games demonstrate the effectiveness of AOP and SIGCAM.

## Acknowledgments

This work is funded by the Natural Science Foundation of China (No.61673204).

## References


Adebayo, J.; Gilmer, J.; Goodfellow, I. J.; and Kim, B. 2018. Local Explanation Methods for Deep Neural Networks Lack Sensitivity to Parameter Values. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*.

Chattopadhyay, A.; Sarkar, A.; Howlader, P.; and 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 2018, Lake Tahoe, NV, USA, March 12-15, 2018*, 839–847. doi:10.1109/WACV.2018.00097. URL https://doi.org/10.1109/WACV.2018.00097.

Fong, R. C.; and Vedaldi, A. 2017. Interpretable Explanations of Black Boxes by Meaningful Perturbation. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 3449–3457. doi:10.1109/ICCV.2017.371. URL https://doi.org/10.1109/ICCV.2017.371.

Goyal, P.; Dollár, P.; Girshick, R. B.; Noordhuis, P.; Wesolowski, L.; Kyrola, A.; Tulloch, A.; Jia, Y.; and He, K. 2017. Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. *CoRR* abs/1706.02677. URL http://arxiv.org/abs/1706.02677.

Kapishnikov, A.; Bolukbasi, T.; Viégas, F. B.; and Terry, M. 2019. XRAI: Better Attributions Through Regions. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, 4947–4956. doi:10.1109/ICCV.2019.00505. URL https://doi.org/10.1109/ICCV.2019.00505.

Marcetic, D.; Soldic, M.; and Ribaric, S. 2017. Hybrid Cascade Model for Face Detection in the Wild Based on Normalized Pixel Difference and a Deep Convolutional Neural Network. In *Computer Analysis of Images and Patterns - 17th International Conference, CAIP 2017, Ystad, Sweden, August 22-24, 2017, Proceedings, Part II*, 379–390.

Petsiuk, V.; Das, A.; and Saenko, K. 2018. RISE: Randomized Input Sampling for Explanation of Black-box Models. In *British Machine Vision Conference 2018, BMVC 2018, Northumbria University, Newcastle, UK, September 3-6, 2018*, 151. URL http://bmvc2018.org/contents/papers/1064.pdf.

Qi, Z.; Khorram, S.; and Li, F. 2020. Visualizing Deep Networks by Optimizing with Integrated Gradients. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA, February 7-12, 2020*, 11890–11898. AAAI Press. URL https://aaai.org/ojs/index.php/AAAI/article/view/6863.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, 1135–1144. doi:10.1145/2939672.2939778. URL https://doi.org/10.1145/2939672.2939778.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 618–626. doi:10.1109/ICCV.2017.74. URL https://doi.org/10.1109/ICCV.2017.74.

Seo, D.; Oh, K.; and Oh, I. 2020. Regional Multi-Scale Approach for Visually Pleasing Explanations of Deep Neural Networks. *IEEE Access* 8: 8572–8582. doi:10.1109/ACCESS.2019.2963055. URL https://doi.org/10.1109/ACCESS.2019.2963055.

Shrikumar, A.; Greenside, P.; and Kundaje, A. 2017. Learning Important Features Through Propagating Activation Differences. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, 3145–3153. URL http://proceedings.mlr.press/v70/shrikumar17a.html.

Shrikumar, A.; Greenside, P.; Shcherbina, A.; and Kundaje, A. 2016. Not Just a Black Box: Learning Important Features Through Propagating Activation Differences. *CoRR* abs/1605.01713. URL http://arxiv.org/abs/1605.01713.

Smilkov, D.; Thorat, N.; Kim, B.; Viégas, F. B.; and Wattenberg, M. 2017. SmoothGrad: removing noise by adding noise. *CoRR* abs/1706.03825. URL http://arxiv.org/abs/1706.03825.

Springenberg, J. T.; Dosovitskiy, A.; Brox, T.; and Riedmiller, M. A. 2015. Striving for Simplicity: The All Convolutional Net. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*. URL http://arxiv.org/abs/1412.6806.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic Attribution for Deep Networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, 3319–3328. URL http://proceedings.mlr.press/v70/sundararajan17a.html.

Yan, C.; Zhang, Q.; Zhao, X.; and Huang, Y. 2017. An Intelligent Field-Aware Factorization Machine Model. In *Database Systems for Advanced Applications - 22nd International Conference, DASFAA 2017, Suzhou, China, March 27-30, 2017, Proceedings, Part I*, 309–323. doi:10.1007/978-3-319-55753-3_20. URL https://doi.org/10.1007/978-3-319-55753-3_20.

Zhang, J.; Bargal, S. A.; Lin, Z.; Brandt, J.; Shen, X.; and Sclaroff, S. 2018. Top-Down Neural Attention by Excitation Backprop. *International Journal of Computer Vision* 126(10): 1084–1102. doi:10.1007/s11263-017-1059-x. URL https://doi.org/10.1007/s11263-017-1059-x.

Zhou, B.; Khosla, A.; Lapedriza, À.; Oliva, A.; and Torralba, A. 2016. Learning Deep Features for Discriminative Localization. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2921–2929. doi:10.1109/CVPR.2016.319. URL https://doi.org/10.1109/CVPR.2016.319.