IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 12 February 2024, accepted 13 April 2024, date of publication 23 April 2024, date of current version 6 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3392853

RESEARCH ARTICLE

Grad++ScoreCAM: Enhancing Visual Explanations of Deep Convolutional Networks Using Incremented Gradient and Score-Weighted Methods

SHAFIULLAH SOOMRO[1], ASIM NIAZ[2], AND KWANG NAM CHOI[2]
[1]Department of Computer Science and Media Technology, Linnaeus University, 3016 Växjö, Sweden
[2]Department of Computer Science and Engineering, Chung-Ang University, Seoul 06974, South Korea

Corresponding author: Kwang Nam Choi (knchoi@cau.ac.kr)

This work was supported by the Ministry of Science and Information and Communication Technology (ICT) and National IT Industry Promotion Agency (NIPA) through High Performance Computing (HPC) Support Project.

ABSTRACT We propose a novel method that combines the strengths of two popular class activation mapping techniques, GradCAM++ and ScoreCAM, to improve the interpretability and localization of convolutional neural networks (CNNs). Our proposed method, called ''Grad++-ScoreCAM'', first utilizes the GradCAM++ algorithm to generate a coarse heatmap of an input image, highlighting the regions of importance for a particular class. Then, we employ the ScoreCAM algorithm to refine the heatmap by incorporating the localization information from the intermediate layers of the network. By combining these two techniques, we can generate more accurate and fine-grained heatmaps that highlight the regions of the input image that are most relevant to the prediction of the CNN. We evaluate our proposed method on a benchmark dataset and demonstrate its superiority over existing methods in terms of accuracy and interpretability. Our method has potential applications in various fields, including medical imaging, object recognition, and natural language processing.

INDEX TERMS Class activation, convolutional neural networks, decision-making, recognition, decision interpretation.

I. INTRODUCTION

Deep Neural Networks (DNNs) can be made more transparent by providing explanations that make it possible for people to understand certain aspects of the inferences made by the model. A common method for attaining this goal is to visualize a particular object of interest based on the significance of input attributes or learning weights. As a convolution neural network (CNN) is an essential part of cutting-edge models for image as well as language processing. Several techniques have concentrated on enhancing the explanations of convolutions and convolutional neural networks. Examples of techniques that are frequently used include Class Activation Map [1], Perturbation [2], and Gradient Visualization [3].

Existing class activation map (CAM) based visual interpretation methods still face limitations, notably in accurately capturing intricate decision boundaries and providing fine-grained insights into the reasoning behind complex neural network decisions. Additionally, these methods may struggle with interpretability in certain scenarios, hindering their effectiveness in fully understanding the nuanced decision-making processes of convolutional neural networks.

Gradient-based techniques leverage the backpropagation algorithm to compute the gradient of a target class concerning the input image. This process aims to identify and emphasize

The associate editor coordinating the review of this manuscript and approving it for publication was Senthil Kumar.

regions within the image that have the most significant impact on the model's prediction. The Saliency Map approach, proposed by Simonyan et al. [3], utilizes the derivative of the target class score as an explanation for the model's decision.

Various other methods have been developed to enhance the interpretability of these gradients. For instance, Adebayo et al. [4], Omeiza et al. [5], Springenberg et al. [6], Sundararajan et al. [7], and Zeiler et al. [8] propose techniques that manipulate and refine the gradient information to improve the quality of visualizations. However, it is noted that the resulting saliency maps can be prone to noise and of relatively low quality, as reported by Omeiza et al. [5].

In contrast to gradient-based methods, perturbation-based techniques alter the original input image and observe the corresponding changes in the model's predictions. Chang et al. [9], Dabkowski et al. [10], Fong [11], Petsiuk et al. [12], Ribeiro et al. [2], and Wagner et al. [13] have proposed such methods. These approaches are particularly useful for determining the sensitivity of the model to perturbations in the input data.

It is important to note that perturbation-based methods often require additional regularization techniques, as highlighted by Fong [11], and can be computationally expensive when aiming to identify minimal regions that significantly affect the model's predictions.

The explanations generated by CAM [1] are visual and focused on a single input. They are created using a linear weighted combination of activation maps obtained from convolutional layers. While CAM is effective at producing localized visual explanations, it requires a global pooling layer [14] and is sensitive to architecture. Grad-CAM [15] and its derivatives, including Grad-CAM++ [16], are designed to generalize CAM for models that lack global pooling layers. Later, Score-weighted Class Activation Mapping (Score-CAM) [17] proposed an explainable AI technique that highlights the most important regions of an image that contribute to a particular prediction made by a deep learning model. It does this by computing a class activation map for the predicted class, which represents the regions of the image that were most important for the prediction. One of the drawbacks of Score-CAM is that it relies on the increase in confidence of the model to determine the important regions of the image. This means that if the model is already highly confident in its prediction, Score-CAM may not be very effective at identifying the most important regions of the image.

The interpretability of deep neural networks is pivotal for understanding their decision-making processes, particularly in image classification. While gradient-based techniques like Grad-CAM++ [16] offer valuable insights into salient image regions, their slow convergence speed and computational intensity, especially in complex architectures, hinder real-time applicability. In contrast, Score-CAM [17] improves localization accuracy but at the cost of increased computational complexity, making it slower and resource-intensive. Motivated by these limitations, we propose a novel interpretation method aiming to combine Grad-CAM++ and Score-CAM strengths while addressing their drawbacks. By integrating the efficiency of Grad-CAM++ and the accuracy of Score-CAM, our approach seeks to provide a comprehensive solution with faster convergence speed and reduced computational overhead, making it well-suited for practical applications requiring real-time interpretability. We assign weights to each pixel of the final convolutional feature map of a CNN based on its gradient with respect to a specific spatial position and obtain a gradient-based confidence score. Our main contributions are listed here:

- Our article presents Grad++ScoreCAM, a visual explanation method that utilizes pixel-wise gradients. This innovative approach effectively combines perturbation-based and CAM-based features, resulting in an easily understandable weight derivation for activation maps. Later, we obtain the final result by taking a linear combination of GRAD++ pixel-wise confidence scores with activation maps.
- Our study involves a quantitative evaluation of the saliency maps generated by Grad++ScoreCAM for recognition tasks, utilizing metrics such as Average Drop/Average Increase. Our results demonstrate that Grad++ScoreCAM is more effective at identifying crucial features.

II. BACKGROUND WORK

A. CAM METHOD

Zhou et al. [1] proposed a class activation mapping method, which is a deep learning technique that visualizes the regions of an image that contribute the most to a particular classification decision. CAM generates a heatmap that highlights the important regions in the input image that influenced the classification decision. This technique is commonly used in computer vision applications such as object detection and image segmentation.

One of the main limitations of CAM is that it requires a pre-trained neural network, and the accuracy of the generated heatmaps depends on the performance of the underlying model. Furthermore, CAM may not work well in cases where the classification decision is based on multiple regions of the image, as it may only highlight the most salient region. Finally, CAM may not be suitable for complex image datasets where the features of interest are not well-defined or are distributed throughout the image.

Suppose we have a model f that includes a global pooling layer, denoted as l, positioned after the last convolution layer, $l-1$, and immediately before the last fully connected layer, $l+1$. In the context of a specific class of interest, c, the CAM explanation is represented as L^c_{CAM}, which can be formally defined as follows:

$$L^c_{CAM} = ReLU\Big(\sum_k \alpha^c_k A^k_{l-1}\Big) \tag{1}$$

where l is for the convolutional layer, A^k_l describes the activation map of the k-th channel and α^c_k denotes the weight

of the $k-th$ neuron after pooling connecting two layers l and $l+1$, defined as:

$$\alpha_k^c = w_l^c, l+1[k] \tag{2}$$

CAM was developed with the intention of utilizing the unique spatial information contained within each activation map, denoted as A_l^k, that corresponds to different regions of the input X. The significance of each channel is determined by the weight assigned to the linear combination of the fully connected layer that follows the global pooling operation. However, in the absence of a global pooling layer or in situations where there are no fully connected layers or multiple fully connected layers, the CAM technique cannot be applied as there would be no definition of the weight coefficient α_k^c.

B. GRAD-CAM METHOD

To overcome the aforementioned issue, Grad-CAM [15] introduces an extension to the definition of α_k^c by using the gradient of the class confidence Y^c with respect to the activation map A_l. This leads to the following definition for Grad-CAM:

$$L_{Grad-CAM}^c = ReLU\big(\sum_k \alpha_k^c A_{l-1}^k\big) \tag{3}$$

where

$$\alpha_k^c = GP\big(\frac{\partial Y^c}{\partial A_l^k}\big) \tag{4}$$

where GP is a global pooling operation. Without requiring any retraining or architectural modification, Grad-CAM can work with any deep CNN where the activation maps α_k^c are differentiable functions of the final Y^c.

Gradient-weighted Class Activation Mapping (GradCAM) is an extension of CAM that highlights the regions of an image that are important for a particular classification decision, using the gradients of the target class with respect to the feature maps of the last convolutional layer.

GradCAM overcomes some of the limitations of CAM by providing more precise and localized heatmaps. It can also be used with any CNN model and is not limited to pre-trained networks. Furthermore, GradCAM can be used for a wider range of tasks, including object detection, image segmentation, and visual question answering.

One of the shortcomings of Grad-CAM is that it struggles to accurately identify the location of objects in an image that contains multiple instances of the same class. This is a critical issue as multiple instances of the same object in an image are frequently encountered in real-world situations. Furthermore, because the approach employs an unweighted average of partial derivatives, the localization can often only correspond to bits and pieces of the object, rather than the entire object. As a result, the user's confidence in the model can be undermined, and Grad-CAM's aim of increasing transparency in deep CNNs can be hindered.

C. GRAD-CAM++ METHOD

This method introduces a new and improved explanation algorithm for CNN architecture called Grad-CAM++ [16]. GradCAM++ is an improved version of GradCAM that provides more accurate and fine-grained visual explanations of deep neural network predictions. GradCAM++ generates visual explanations by computing weighted combinations of the final convolutional layer's feature maps. These weights are derived from the gradient information flowing into the convolutional layer. By computing the first-order and second-order gradients of the target class, GradCAM++ can capture more fine-grained localization information and produce more precise heatmaps.

This method proposed a solution to the issues discussed earlier and calculated a weighted average of the pixel-wise gradients. The proposed formulation of this method is defined as:

$$L_{Grad-CAM++}^c = \sum_k \alpha_k^c . relu\big(\frac{\partial Y^c}{\partial A_l^k}\big) \tag{5}$$

where

$$\alpha_k^c = \big(\frac{\partial Y^c}{\partial A_l^k}\big) \tag{6}$$

In this context, the α_k^c values represent the coefficients used to weight the pixel-wise gradients for class c and convolutional feature map A_l^k. By utilizing these coefficients, all feature maps receive equal emphasis in identifying the presence of objects. The formulation for Grad-CAM is a subset of GradCAM++. As a result, GradCAM++ can be thought of as a generalized version of Grad-CAM, as implied by its name.

While GradCAM++ has improved on the limitations of the original GradCAM, it still has some limitations of its own. It may struggle with identifying objects or features that are small or occluded, and it may produce heatmaps that are difficult to interpret or contain false positives.

D. SCORE-CAM METHOD

Unlike earlier approaches [15], [16], that rely on the gradient information from the last convolutional layer to indicate the significance of each activation map, Score-CAM [17] considers the Increase of Confidence as the measure of importance. ScoreCAM generates a weight matrix by computing the channel-wise scores of the intermediate activation maps. It then multiplies this weight matrix with the activation maps to obtain a class-discriminative localization map. This map highlights the regions that are important for the classification decision.

Consider a function $Y = f(X)$ that accepts an input vector $X = [x_0, x_1, \ldots, x_n]^T$ and returns a scalar Y. The contribution c_i of x_i where $i \in [0, n-1]$ towards Y can be determined by replacing the i-th entry in the known baseline input X_b with x_i and observing the resulting change in the output. This can be expressed formally as:

$$c_i = f\left(X_b \circ H_i\right) - f\left(X_b\right) \tag{7}$$

where we define the vector H_i as having the same shape as X_b. For Each entry h_j in H_i, $h_j = \mathbb{I}[i = j]$. Finally, the Hadamard product is denoted by $\circ$. Defining Channel-wise Increase of Confidence as a metric, this method quantifies the significance of each activation map by utilizing it to enhance the confidence level. This idea is defined as follows:

We have a CNN model, represented as $Y = f(X)$, which accepts an input X and generates a scalar output Y. To calculate the contribution of a specific convolutional layer l and its corresponding activation A, we select the k-th channel of A_l as A_l^k. Given a known baseline input X_b, we define the contribution of A_l^k towards Y as follows:

$$C(A_l^k) = f\left(X \circ H_l^k\right) - f\left(X_b\right) \tag{8}$$

where

$$H_l^k = s\left(Up(A_l^k)\right) \tag{9}$$

We use Up$(\cdot)$ to represent the process of upsampling A_l^k to match the input size. Furthermore, $s()$ denotes a normalization function that maps each element of the input matrix to a value within the range of [0, 1].

Finally, referring to the notations used earlier, if we focus on a specific class of interest denoted by c, and a convolutional layer l within the model f, then the Score-CAM $L^c_{Score-CAM}$ can be defined as follows:

$$L^c_{Score-CAM} = ReLU\left(\sum_k \alpha_k^c A_{l-1}^k\right) \tag{10}$$

where

$$\alpha_k^c = C\left(A_l^k\right) \tag{11}$$

In the previously defined expression for Score-CAM $L^c_{Score-CAM}$, the function $C()$ represents the CIC score for the activation map A_l^k.

III. FORMULATING ENHANCED VISUAL EXPLANATIONS OF DEEP CONVOLUTIONAL NETWORKS USING INCREMENTED GRADIENT AND SCORE-WEIGHTED METHODS

The proposed method involves combining two techniques, Grad++ [16] and Score-CAM [17], to generate a heatmap that highlights the most important regions of an input image with respect to a particular class. Grad++ is a technique used to compute the gradients of the class score Y^c with respect to the input image. This provides information about the areas of the image that contribute most to the classification decision.

Score-CAM, in contrast, generates a class-discriminative localization map by weighting the activation maps of a convolutional layer based on the class score. This technique allows for the identification of the regions of the image that are most important for the prediction of a particular class.

By combining Grad++ and Score-CAM, the proposed method generates a heatmap that highlights the most important regions of an input image for a particular class.

Algorithm 1 Grad++ScoreCAM

Require:
Image X
CNN model f
Class of interest c
Baseline input X_b

Ensure:
Normalized heatmap matrix H'
Smoother masks H_L^k

1: Compute the gradients of the class score Y_c with respect to the input image.
2: Select an internal convolutional layer l within f and obtain the corresponding gradient activation A_l.
3: **for** each k-th channel of A_l **do**
4: Compute the contribution $C(A_l^k)$:

$$C(A_l^k) = f(X \otimes H_l^k) - f(X_b)$$

5: Compute normalized activation maps H_L^k using the min-max normalization function $s()$:

$$H_L^k = s(A_l^k) = \frac{A_l^k - \min(A_l^k)}{\max(A_l^k) - \min(A_l^k)}$$

6: **end for**
7: Calculate Grad++ScoreCAM for the convolutional layer l and class of interest c:

$$L_c^{\text{Grad++ScoreCAM}} = \text{ReLU}\left(\sum_k (\omega_{ck} \cdot A_{l-1}^k)\right)$$

8: Resize the original input image to match the size of activation weights.
9: Perturb the resized input image with the gradient activation maps and evaluate the significance of each activation map:

$$\text{Target Score} = f(X \otimes H_l^k)$$

10: Normalize the resulting heatmap matrix H using the min-max normalization function $s()$:
$h'_{ij} = \frac{h_{ij} - \min(H)}{\max(H) - \min(H)}$.

The Grad++ technique provides information about the importance of individual pixels, while the Score-CAM method captures the importance of features at a higher level. Together, these two techniques provide a more comprehensive understanding of the image features that contribute to a specific classification decision. Fig 1 depicts the sequence of steps in detail.

Unlike earlier approaches [15], [16], that rely on the gradient information from the last convolutional layer to indicate the significance of each activation map, Grad++ScoreCAM also relies on gradient-based weights. The proposed method, however, considers gradient weights as a measure of importance based on their ability to increase confidence in the classification decision. Consider a CNN model $Y = f(X)$

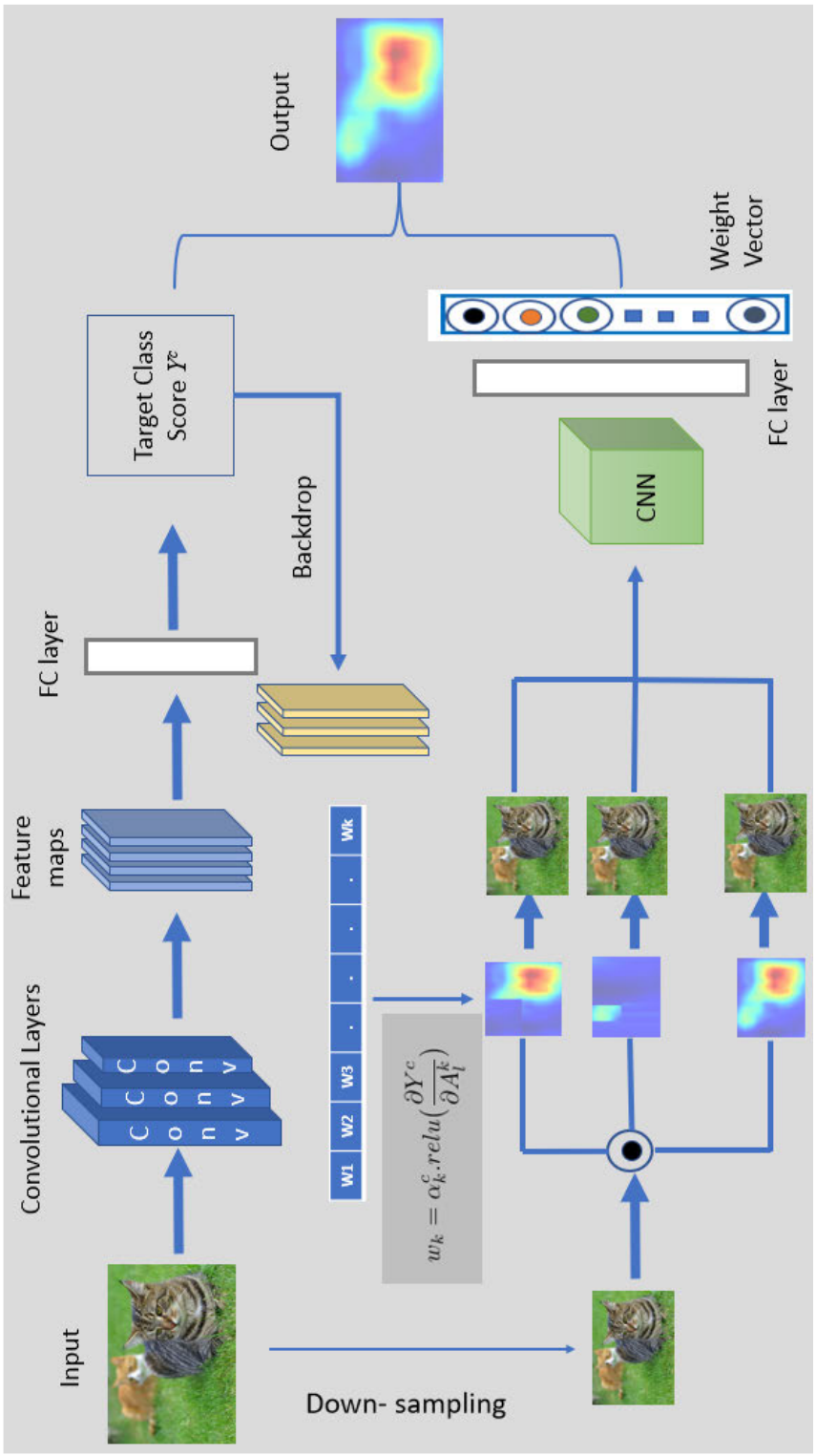

FIGURE 1. Our suggested Grad++-ScoreCAM functions via a pipeline with two steps. Phase 1 involves the extraction of weighted gradients. The result is a forward-passing score on the target class for each weight, which acts as a mask on the original image. This procedure is repeated k times in Phase 2, where k is the number of weights. A linear combination of class scores and gradient weights yields the final outcome.

that takes an input X and produces a scalar Y as the output. Let us select an internal convolutional layer l within f and the corresponding gradient activation A. We can denote the k-th channel of A_l as A_l^k. Given a known baseline input X_b, we define the contribution of A_l^k towards Y as follows:

$$C(A_l^k) = f\left(X \circ H_l^k\right) - f\left(X_b\right) \tag{12}$$

where

$$H_l^k = s\left(A_l^k\right) \tag{13}$$

$s()$ denotes a normalization function that maps each element of the input matrix to a value within the range [0, 1].

Finally, assuming the notation presented in earlier sections, let us define Grad++-ScoreCAM $L_{Grad++ScoreCAM}$ as follows for a convolutional layer l within a model f and a given class of interest c.

$$L^c_{Grad++Score-CAM} = ReLU\left(\sum_k \alpha_k^c A_{l-1}^k\right) \tag{14}$$

where

$$\alpha_k^c = C\left(w_k^c\right) \tag{15}$$

where the function $C()$ represents the score for the weights w_k^c.

Unlike Score-Cam [17] we downsize the original input image corresponding to the size of activation weights we get. Subsequently, we perturb the resized input with the gradient activation maps. The significance of this activation map is obtained by evaluating the target score of the perturbed input. In this way, each activation map not only indicates the spatial locations that are most relevant to an internal activation map but can also directly act as a mask to perturb the input image.

When using Increase of Confidence, a binary mask H_i is created on top of the input, preserving only the feature of interest in the input. However, this binary mask may not be appropriate if we are interested in a specific region of the input image rather than a single pixel. Therefore, to generate a smoother mask H_L^k for an activation map, we normalize the raw activation values in each activation map to a range of [0, 1]. Instead of assigning binary values to all elements, we use the normalization function as:

$$s(A_l^k) = \frac{A_l^k - minA_l^k}{maxA_l^k - minA_l^k} \tag{16}$$

A comprehensive account of the implementation can be found in the Algorithm 1.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed explanation method's effectiveness is assessed through experiments in this section. Our approach is first evaluated qualitatively using visualization on ImageNet. In the following experiments, the Resnet, VGG16 and AlexNet pretrained models are used as the base models. Moreover, we use the publicly available object classification dataset, ILSVRC2012 [18] for our experiments. The input images are resized to (224 × 224 × 3), transformed to the range [0, 1], and normalized using the mean vector [0.485, 0.456, 0.406] and standard deviation vector [0.229, 0.224, 0.225].

Moreover, during the computation phase of Score-CAM, we employ a different approach to handle the activation maps. Rather than enlarging the activation maps to match the resolution of the input image, we choose to reduce the input image to the resolution of the activation maps. This technique effectively minimizes the computational workload and, consequently, reduces the overall time complexity involved.

The study involves a qualitative comparison of saliency maps produced using five state-of-the-art methods: gradient-based, perturbation, and CAM-based. The proposed method generates saliency maps that are visually more understandable and contain fewer random noises. Fig 2 illustrates the results of the proposed idea based on the pre-trained Resnet model and compared to CAM [1], GradCAM [15], GradCAM++ [16], ScoreCAM [17] and SmoothGrad-CAM [5]. The results demonstrate that our approach produces fewer random noises. Furthermore, our approach creates smoother saliency maps.

As the class activation map is model-agnostic, we sought to evaluate the generalizability of our proposed method by conducting experiments using other pre-trained deep neural network models, namely VGG16 and Alexnet, as illustrated in Figure 3 and Figure 4. Our visual analysis demonstrates that the proposed method exhibits exceptional performance, meeting or exceeding current state-of-the-art techniques.

A. EVALUATING FAITHFULNESS USING IMAGE RECOGNITION

To evaluate the faithfulness of the explanations generated by the proposed method for object recognition, we follow the approach described in [16]. This involves masking the original input with saliency maps and observing the resulting score change on the target class. We use the same metrics as [16] to measure the quality of the results. The Average Drop is computed as $\sum_{i=1}^{N} \frac{max(0, Y_i^c - O_i^c)}{Y_i^c} X100$. Likewise, the Increase in Confidence (also known as Average Increase) is expressed as $\sum_{i=1}^{N} \frac{Sign(Y_i^c < O_i^c)}{N}$. In this context, Y_i^c represents the predicted score for class c for a given image i, while O_i^c refers to the predicted score for class c obtained when the explanation map region is used as input where Sign returns 1 if the input is true. We conduct the experiment on the ImageNet ILSVRC2012 [18] validation set, randomly selecting 1000 images. Our results are presented in Table1.

Table1 demonstrates that the proposed method achieves an average drop of 32.6 percent and an average increase of 31.4 percent. These results are vastly superior to those of other state-of-the-art existing methods. The recognition task performance indicates that the proposed approach can identify the most distinctive region of the target object, instead of relying on subjective human judgment. We have also compared the results with those of gradient-based methods

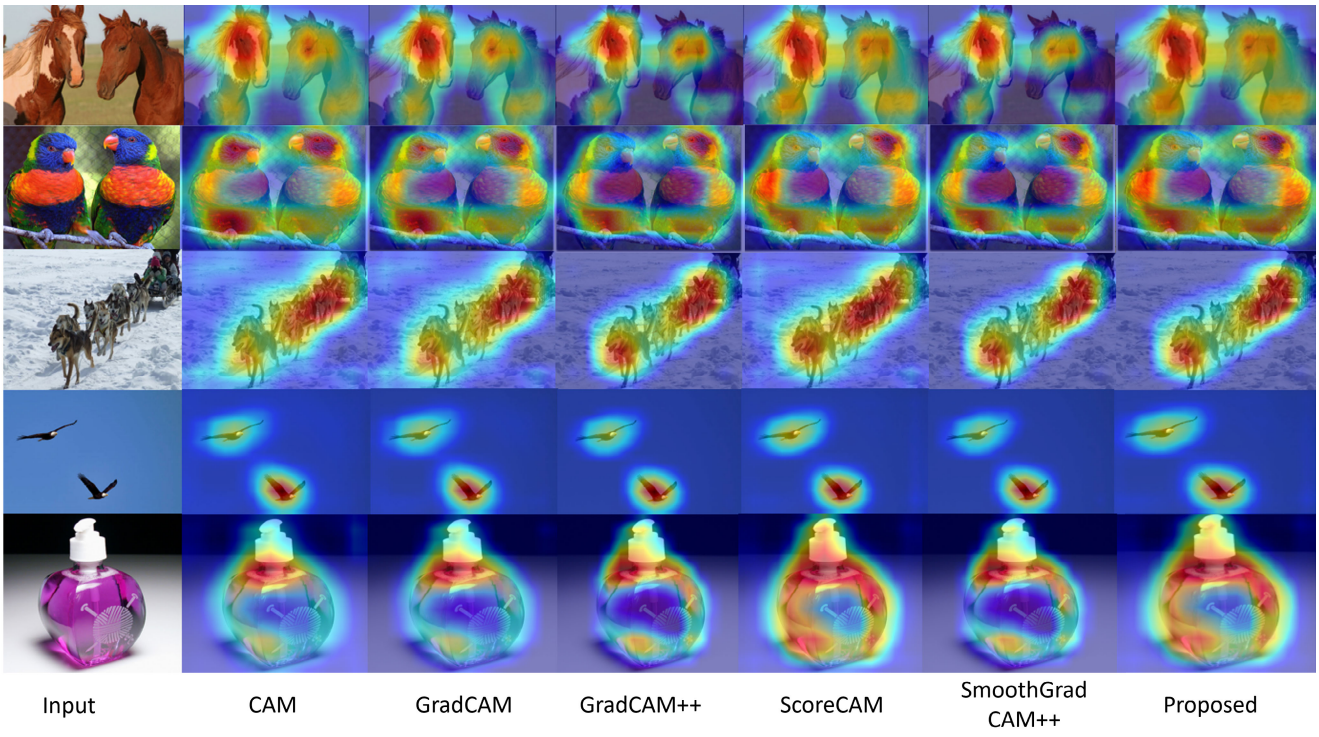

FIGURE 2. Visualization of CAM [1], GradCAM [15], GradCAM++ [16], ScoreCAM [17], SmoothGradCAM [5] and proposed method using pre-trained Resnet model over ILSVRC2012 [18] dataset.

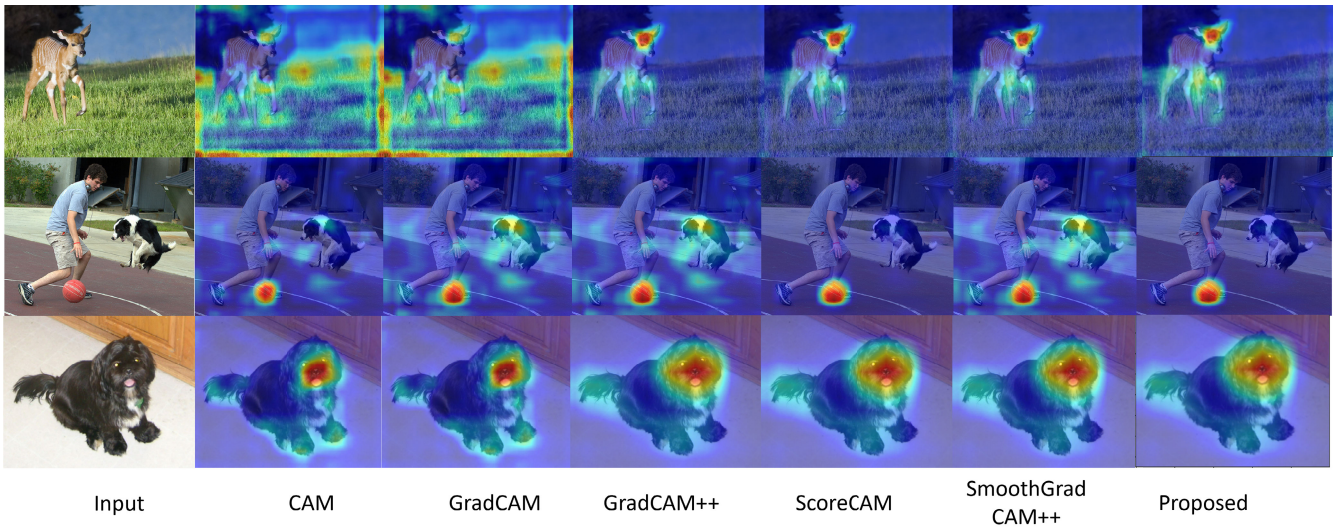

FIGURE 3. Visualization of CAM [1], GradCAM [15], GradCAM++ [16], ScoreCAM [17], SmoothGradCAM [5] and proposed method using pre-trained VGG16 model over ILSVRC2012 [18] dataset.

TABLE 1. Experimental conditions.

Methods	CAM	GradCAM	GradCAM++	ScoreCAM	SmoothGradCAM	Proposed
Average Drop %	47.2	47.8	45.5	31.5	29.1	32.6
Average Increase %	13.0	19.6	18.9	30.6	30.1	31.4

owing to their similar visual characteristics. The recognition task findings suggest that the proposed method provides a more accurate representation of the decision-making process of the original CNN model compared to previous approaches.

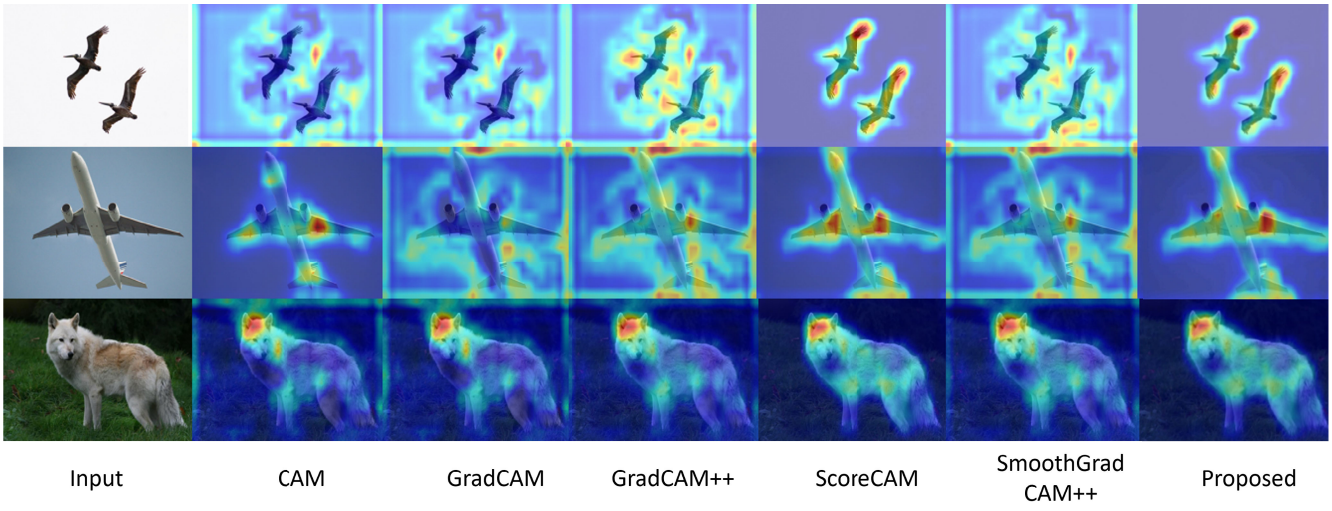

FIGURE 4. Visualization of CAM [1], GradCAM [15], GradCAM++ [16], ScoreCAM [17], SmoothGradCAM [5] and proposed method using pre-trained Alexnet model over ILSVRC2012 [18] dataset.

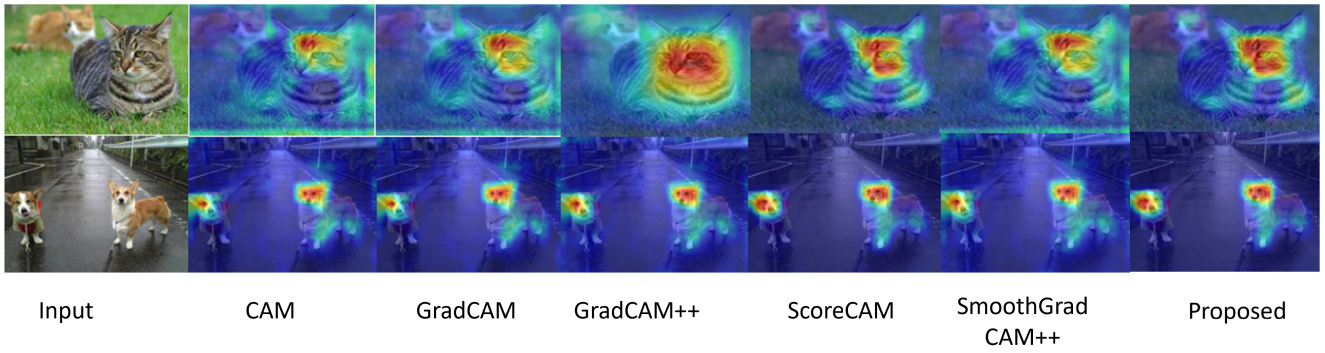

FIGURE 5. Multi object Visualization of CAM [1], GradCAM [15], GradCAM++ [16], ScoreCAM [17], SmoothGradCAM [5] and Proposed methods using pre-trained Alexnet model over ILSVRC2012 [18] dataset.

B. MULTI-OBJECT VISUALIZATION

Not only is the proposed method adept at accurately localizing a single object, it also surpasses prior techniques in detecting multiple objects of the same class, as demonstrated in the results presented in Figure 5. While CAM and Grad-CAM typically only identify a single object in an image, both Grad-CAM++ and Score-CAM are capable of locating multiple objects. However, the proposed approach and ScoreCAM excel in generating saliency maps that are highly focused, surpassing those of previous methods. This is attributed to the fact that the weight of each activation map in the proposed work is determined by its gradient score on the target class, allowing for independent focus on each target object with high confidence. As a result, all relevant pieces of evidence related to the target class can be identified and combined via a linear combination.

V. CONCLUSION

We obtain the initial localization map, which is then passed to the second phase of ScoreCAM to generate the final saliency map. We found that our model outperformed GradCAM++ as well as ScoreCAM in terms of accuracy and robustness when evaluated using various network models on the Imagenet dataset.

By leveraging the complementary strengths of GradCAM++ and ScoreCAM, our GRADScoreCAM++ model was able to produce more accurate saliency maps, particularly in cases where multiple objects of the same class are present in an image. This was achieved by incorporating the ability of ScoreCAM to locate multiple objects while maintaining the high-resolution and focused saliency maps of GradCAM++. Our evaluation results demonstrate that the proposed GRADScoreCAM++ model is not only effective but also robust, as it consistently outperformed both GradCAM++ and ScoreCAM across diverse network models. These findings suggest that our approach can be generalized to other visual explanation methods to develop more accurate and robust models for a wide range of applications. This model has been evaluated on the ImageNet dataset and has shown promising results, demonstrating

the potential of our approach for a wide range of applications.

ACKNOWLEDGMENT

(Shafiullah Soomro and Asim Niaz are co-first authors.)

REFERENCES

[1] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2921–2929.

[2] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you? Explaining the predictions of any classifier," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1135–1144.

[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[4] J. Adebayo, J. Gilmer, I. Goodfellow, and B. Kim, "Local explanation methods for deep neural networks lack sensitivity to parameter values," 2018, *arXiv:1810.03307*.

[5] D. Omeiza, S. Speakman, C. Cintas, and K. Weldermariam, "Smooth grad-CAM++: An enhanced inference level visualization technique for deep convolutional neural network models," 2019, *arXiv:1908.01224*.

[6] J. Tobias Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," 2014, *arXiv:1412.6806*.

[7] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proc. Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 3319–3328.

[8] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.* Zurich, Switzerland: Springer, 2014, pp. 818–833.

[9] C.-H. Chang, E. Creager, A. Goldenberg, and D. Duvenaud, "Explaining image classifiers by counterfactual generation," 2018, *arXiv:1807.08024*.

[10] P. Dabkowski and Y. Gal, "Real time image saliency for black box classifiers," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–10.

[11] R. C. Fong and A. Vedaldi, "Interpretable explanations of black boxes by meaningful perturbation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3449–3457.

[12] V. Petsiuk, A. Das, and K. Saenko, "RISE: Randomized input sampling for explanation of black-box models," 2018, *arXiv:1806.07421*.

[13] J. Wagner, J. M. Köhler, T. Gindele, L. Hetzel, J. T. Wiedemer, and S. Behnke, "Interpretable and fine-grained visual explanations for convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9089–9099.

[14] M. Lin, Q. Chen, and S. Yan, "Network in network," 2013, *arXiv:1312.4400*.

[15] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.

[16] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 839–847.

[17] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu, "Score-CAM: Score-weighted visual explanations for convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 111–119.

[18] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.