applied sciences

MDPI

Article

Attention Map-Guided Visual Explanations for Deep Neural Networks

Junkang An and Inwhee Joe *

Department of Computer Software, Hanyang University, Seoul 04763, Korea; junhang@hanyang.ac.kr
* Correspondence: iwjoe@hanyang.ac.kr; Tel.: +82-02-2220-1088

Abstract: Deep neural network models perform well in a variety of domains, such as computer vision, recommender systems, natural language processing, and defect detection. In contrast, in areas such as healthcare, finance, and defense, deep neural network models, due to their lack of explainability, are not trusted by users. In this paper, we focus on attention-map-guided visual explanations for deep neural networks. We employ an attention mechanism to find the most important region of an input image. The Grad-CAM method is used to extract the feature map for deep neural networks, and then the attention mechanism is used to extract the high-level attention maps. The attention map, which highlights the important region in the image for the target class, can be seen as a visual explanation of a deep neural network. We evaluate our method using two common metrics: average drop and percentage increase. For a more effective experiment, we also propose a new metric to evaluate our method. The experiments were carried out to show that the proposed method works better than the state-of-the-art explainable artificial intelligence method. Our approach can provide a lower average drop and higher percent increase when compared to other methods and find a more explanatory region, especially in the first twenty percent region of the input image.

Keywords: explainable artificial intelligence; visual explanation; attention mechanism

Citation: An, J.; Joe, I. Attention Map-Guided Visual Explanations for Deep Neural Networks. *Appl. Sci.* **2022**, *12*, 3846. https://doi.org/10.3390/app12083846

Academic Editor: Agostino Forestiero

Received: 16 March 2022
Accepted: 10 April 2022
Published: 11 April 2022

1. Introduction

Deep neural networks (DNNs) have enabled tremendous improvements in a number of computer vision tasks, such as image classification [1,2], object detection [3–5], and semantic segmentation [6]; and in some other tasks, such as visual question answering [7] and autonomous driving [8]. However, DNNs are difficult to analyze and behave as black boxes. When designing a deep neural network model, most researchers emphasize the model's framework and the many internal parameters of the model, but they cannot provide a correct explanation of the model's output when the model makes mistakes. This also makes users unable to trust the network's decisions in industries such as healthcare, finance, and security. It is important that we construct transparent models so that they can show users their reasoning. This will help with understanding failures, debugging, and identifying potential biases in training data.

To solve these problems, explainable artificial intelligence (XAI) technology has been proposed, and more and more researchers are working on this technology every year. XAI technology focuses on how to make a DNN model's decisions more transparent, understandable, and trustworthy to humans. To interpret a deep neural network model, it would be useful to generate an explanation map that highlights important regions that are most related to the model's decision. One common approach for interpreting deep neural network models is relying on the changes in the model output, such as the changes in prediction scores concerning the input images [9]. RISE [10] advocated a general approach that probes the model with randomly masked versions of the image and obtains the corresponding outputs without requiring access to its internals for each network architecture. LIME [11] draws random samples and builds an approximated linear decision

model to interpret deep neural networks. However, it depends on super-pixels, which may or may not capture the relevant areas. Another approach, Grad-CAM [12], relies on the gradients by back-propagating the prediction score through the last convolutional layer and applying them as weights to combine the forward feature maps to produce explanations. However, an explanation using Grad-CAM has too much meaningless information, since the feature maps are not necessarily related to the target class.

In this paper, we propose an attention-map-guided visual explanation method for deep neural networks. We use an attention mechanism to generate the attention map from the feature map, which is generated using Grad-CAM. Herein, we compare our approach with other state-of-the-art XAI methods. We evaluate our method using three metrics, and the experimental results show that our method can provide a better explanation than the other methods. Figure 1 shows an overview of our methods. In experiments, we demonstrated the effectiveness of our method using the Imagenet dataset. Our method found the most important region for the deep neural network. Our methodology achieved a lower average drop and a higher percent increase, and uncovered a more explanatory region.

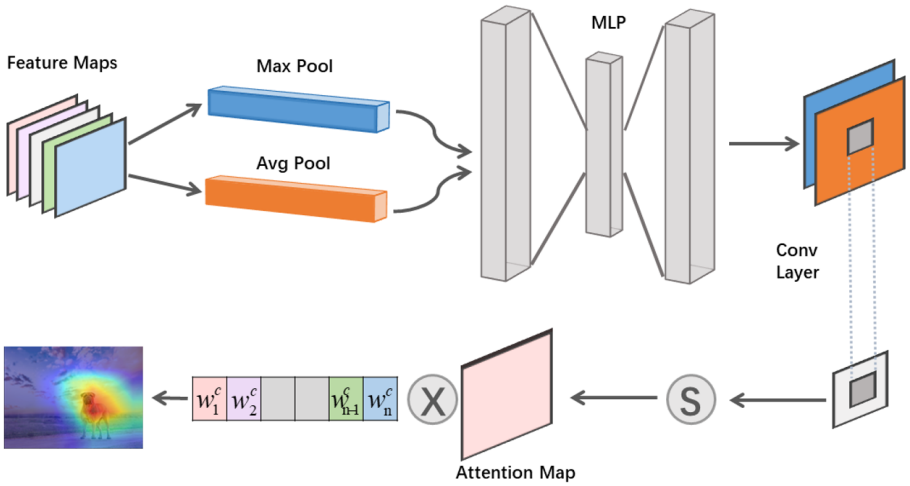

Figure 1. An overview of attention map-guided visual explanations for deep neural networks.

2. Related Work

2.1. CAM-Based XAI Methods

There are now many ways of using class activation mapping (CAM) [13] based methods for explaining the output of a model. These XAI methods use CAM methods as their basis, and some researchers upgrade CAM methods with a mix of backpropagation gradients and feature maps of a certain convolutional layer to generate an explanation map. To generate the explanation map, they have mainly used prior position information, such as part-level bounding boxes and segmentation masks [14]. The CAM is essentially a weighted linear sum of these visual patterns' existence in various spatial regions. It can determine the images' most important regions for the given category by simply upsampling the class activation map to the size of the input image. In Figure 2, the global average pooling (GAP) layer is used to convert the feature map into a feature vector, and each layer of the feature map can be represented as a numerical value. CAM methods multiply the weights corresponding to the bull mastiff class by the layers corresponding to the feature map, making a weighted linear sum. Using a CAM method, it is possible to observe which area the model is looking at. However, the CAM method has some shortcomings; e.g., it needs to change the model's structure from a fully connected layer to a global average pooling layer. Users are cautious to explain DNN models using the CAM technique, since it requires changing the model's basic structure. Changing the model's internal structure is not convenient for the user.

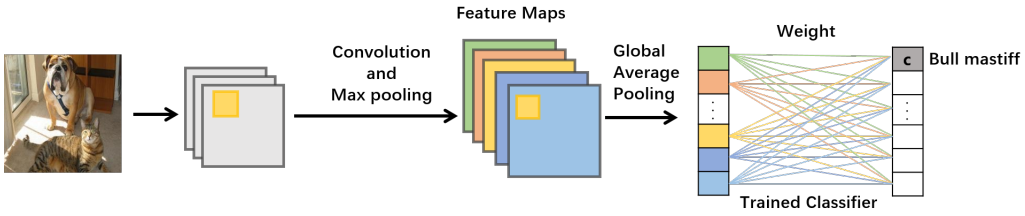

Figure 2. The framework of the CAM method. A series of feature maps are obtained by a forward propagation; then, a global average pooling layer and a trained classifier are used to obtain the output. The CAM method obtains a class activation map by multiplying these weights with the feature maps.

To address these problems of CAM methods, Selvaraju et al. [12] proposed using gradient calculations instead of GAP. Grad-CAM is a new method for combining feature maps using the gradient weights without any modifications to the network structure. It allows any gradients to flow into the final convolutional layer to build a coarse localization map that highlights the regions essential in the image for the predicting class. Grad-CAM assigns priority values to each neuron for a specific choice using the gradient information flowing into the last convolutional layer of the CNN model.

CAM and Grad-CAM use a linear combination of activation to produce a fine-grained explanation. Grad-CAM++ is a Grad-CAM enhancement that provides a visual explanation for the associated class by using a weighted mixture of the positive partial derivatives of the target layers' feature maps concerning a predetermined class score as weights. To create an enhanced visual explanation of multiple objects in a single image, SmoothGrad-CAM was created [15], which is a simple method that can help visually sharpen gradient-based sensitivity maps. Additionally, it can visually brighten gradient-based sensitivity maps, which obtain random samples in the neighbor of an input x and average the sensitivity maps. The gradient of the class score function for the input image is a good starting point for SmoothGrad-CAM. Omeiza, D et al. [16] proposed SmoothGrad-CAM++, which combines SmoothGrad-CAM and Grad-CAM++. Smooth Grad-CAM++ creates visual explanations of the input images that are more visually sharp. Smooth Grad-CAM++ allows one to visualize a layer, a subset of feature maps, or a subset of neurons inside a feature map at each occurrence. Although these XAI methods can provide reasonable visualizations, the majority of them lack obvious and sufficient theoretical backing. XGrad-CAM [17] was proposed to satisfy those needs as much as is feasible, and the studies on it show that ot is a more sensitive and conservation-oriented variant of Grad-CAM. However, because the feature maps are not always connected to the target class, the outputs of activation-based approaches may collect too much worthless information.

2.2. Attention-Based Methods

Attention mechanisms are widely used in the field of natural language processing (NLP) as a way to improve the performances of models [18,19]. They have been employed extensively in sequential models using recurrent neural networks and long short-term memory (LSTM). Evermore research is applying attention mechanisms to computer vision tasks [20,21]. Researchers can use an attention method to extract high-level features to improve the performance of a deep learning model. An attention mechanism in computer vision tasks can be thought of as a dynamic selection process that is implemented by adaptively weighting characteristics based on their relevance to the input. In the past few years, researchers have found that focusing the attention mechanism on many image recognition tasks can provide good results. Some created a global-and-local attention (GALA) module and incorporated it into a DNN model, and the experimental results show the module can improve visual recognition performance [22]. Increasingly, the attention mechanism is being used in the XAI field. The authors of [23,24] offer spatial attention maps of visual sections that the network attends to, which can be shown in a user-friendly manner. However, attention maps are only one element of the puzzle. Non-salient picture

content is filtered away using the attention technique. Attention networks, on the other hand, must locate all potentially salient visual areas and forward them to the primary recognition network for a final decision, just as a human would utilize peripheral vision to determine that "something is there" before visual fixating on the item to determine what it is Kim et al. [25] used a visual attention model that highlights image regions that potentially influence a network's output then applies a causal filtering step to determine which input regions actually influence the output. This produces more succinct visual explanations and more accurately exposes the network's behavior than do other methods. Their research first showed that training with attention does not degrade the performance of the end-to-end network. However, they used a convolutional feature extractor to directly extract the low-level feature map from the image. Thus, the explanation of the deep learning model is based on another deep learning model, and whether the low-level features extracted directly from the input image are the same.

3. Methods

3.1. Grad-CAM

Grad-CAM uses gradient calculations instead of GAP. As shown in Figure 3, Grad-CAM is a method for combining feature maps using gradient weights without any modifications to the network structure. It allows any gradients to flow into the final convolutional layer to build an explanation map that highlights the regions essential in the image for predicting the class. We found through our experiments that using Grad-CAM as a base gave the best results, so we built on it for our subsequent research.

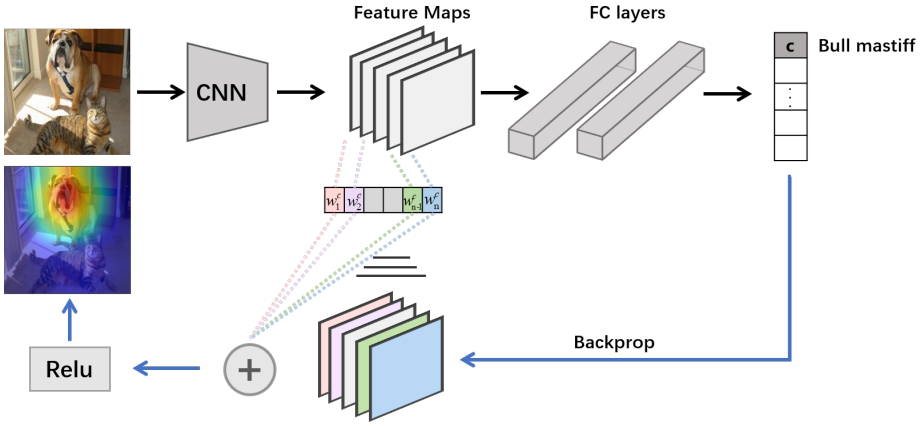

Figure 3. Grad-CAM overview. The input image is processed by the CNN model, and a raw score for the specific class is obtained. The gradients were set to 0, and the "bull mastiff" class was set to 1. Then, it back-propagates the gradient to the rectified convolutional feature maps, which were combined to produce the coarse red heat map that depicts where the model looking.

The Grad-CAM technique computes the gradient of the class score y^c with respect to the feature map of the last convolution layer:

$$\frac{\partial y^c}{\partial A_{ij}^k} \tag{1}$$

It uses global-average-pooling gradients to get weights W_k^c.

$$W_k^c = \frac{1}{Z}\sum_i\sum_j \frac{\partial y^c}{\partial A_{ij}^k} \tag{2}$$

Grad-CAM generalizes visual explanations using a weighted combination of feature maps with ReLU.

$$L^c_{\text{Grad-CAM}} = \text{ReLU}\left(\sum_k W^c_k A^k\right) \tag{3}$$

In Equation (4), weight α^c_k represents a partial linearization of the deep network downstream from A, Z is the total number of feature map cells, y^c is an activation class score for class c, and A^k_{ij} represents activation of the cell at spatial location i.

$$\alpha^c_k = \frac{1}{Z}\sum_i \sum_j \frac{\partial y^c}{\partial A^k_{ij}} \tag{4}$$

Grad-CAM assigns priority values to each neuron for a specific choice using the gradient information flowing into the last convolutional layer of the CNN model.

3.2. Attention Mechanism

3.2.1. General Form

When we become aware of a scene in our lives, we focus our attention on discriminative areas and process them quickly, and almost all existing attention mechanisms can be summed up by Equation (5). $g(x)$ reflects the process of attending to discriminative regions, which corresponds to the process of providing attention. Here, $f(g(x), x)$ denotes that input x is processed based on the attention $g(x)$, which is compatible with processing crucial sections and obtaining information.

$$Attention = f(g(x), x) \tag{5}$$

3.2.2. Channel–Spatial Attention Module

Inspired by Woo et al. [26], we designed our channel–spatial attention module. Distinct channels in different feature maps typically represent different objects in deep neural networks [27]. Channel attention adjusts the weight of each channel as needed, and can be thought of as an objective selection process that determines what to pay attention to. By utilizing the inter-channel relationship of features, we create a channel attention map, wherein each channel of the feature map acts as a feature detector. As shown in Figure 4, we aggregate the spatial information of a feature map by using both average-pooling and max-pooling operations, thereby generating average-pooled feature $AvgPool(F)$ and max-pooled feature $MaxPool(F)$. Both descriptors are then forwarded to a multi-layer perceptron (*MLP*) to produce a channel attention map *Mc*. In short, the channel attention is computed as in Equation (6).

$$Mc(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \tag{6}$$

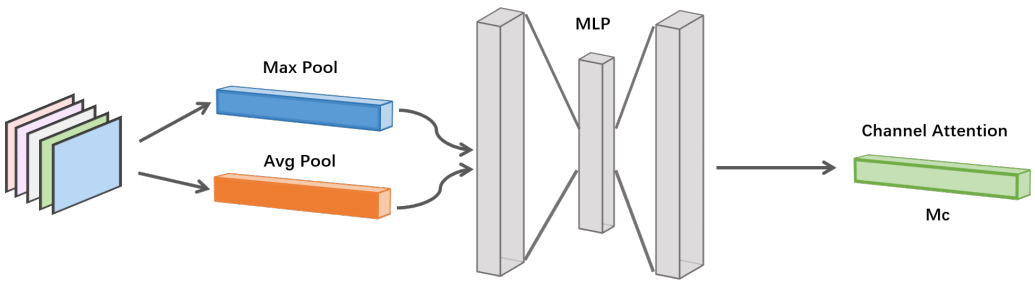

Figure 4. The overview of the channel-attention module, which utilizes both max-pooling outputs and average-pooling outputs with an *MLP*.

We created a spatial attention module that is distinct from channel attention in that it focuses on where there is an informative component, which is complementary to channel attention. As shown in Figure 5, we use average-pooling and max-pooling procedures along the channel axis, and then we use a convolution layer to generate a spatial attention map.

Pooling procedures along the channel axis have been shown to help identify informative regions [28]. We use two pooling operations: average-pooled features $AvgPool(F)$ and max-pooled features $MaxPool(F)$. After that, a convolution layer convolves them to generate our 2D spatial attention map. In Equation (7), σ denotes the sigmoid function, and $f^{7\times7}$ represents a convolution operation with the filter size 7×7. The benefit of the channel–spatial attention module is that it can adaptively identify essential objects and regions. Our attention module leverages both channel and spatial relationships of features to instruct the network on what to focus on and where to focus by sequentially combining channel and spatial attention. It highlights helpful channels while also increasing informative local locations.

$$Ms(F) = \sigma(f^{7\times7}([AvgPool(F); MaxPool(F)])) \tag{7}$$

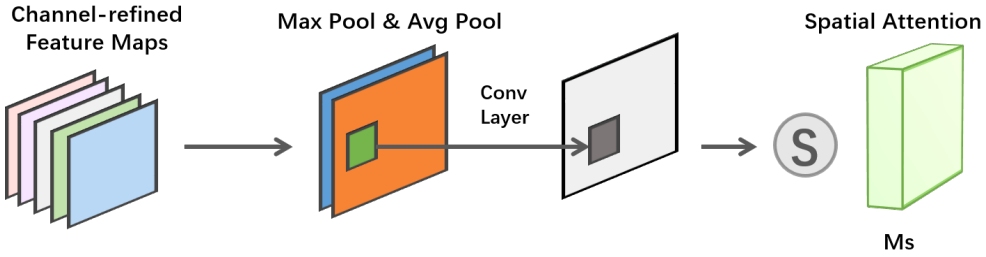

Figure 5. The spatial attention module pools two outputs along the channel axis and sends them to a convolution layer.

4. Experiments

4.1. Experimental Setup

Our experiments were conducted on the commonly-used computer vision dataset ImageNet. They involved the objective evaluation of our method and its compared with Grad-CAM, Grad-CAM++, XGrad-CAM, and SmoothGrad-CAM++. We first tested VGG19 [29], Resnet-50 [30], and Googlenet [31] models, which are pre-trained on ImageNet. After the test, we chose the best-performing model as our black-box model to be explained. All datasets were resized to $3 \times 224 \times 224$ pixels, then transformed to tensors, and finally, normalized to the range [0, 1]. As shown in Table 1, AMD Ryzen 7 3700X was used as the CPU, and a total of 64 GB of memory was used. We used the GeForce RTX 2080 Ti as the GPU. We also used Python 3.6, Pytorch 1.8.1, Torchvision 0.9.1, and other libraries as our environment.

Table 1. Workstation configuration.

Software or Hardware	Specification
CPU	AMD Ryzen 7 3700X
GPU	GeForce RTX 2080 Ti
RAM	DDR4 64 GB
Python	3.6
Pytorch	1.8.1
Torchvision	0.9.1

First, we tested the above pre-trained models and selected the best-performing model for the following experiments. According to Table 2, the Resnet-50 model performs best on the ImageNet dataset, so we chose the Resnet-50 model as the black-box model to be explained.

Table 2. The performance of Resnet-50, Googlenet, and Vgg19.

	Resnet-50	Googlenet	Vgg19
Mean accuracy	0.9496	0.9363	0.9372
top5-error	5.38	8.26	9.21

4.2. Evaluation Metrics

We leveraged the study presented in [32] for the objective evaluation of our proposed method. A heatmap was created for each image using a visualization approach such as Grad-CAM. The most relevant discriminative regions were highlighted in red on this heat map. The primary concept behind a heat map is to create an image that only contains the sub-regions of the original image that are highlighted using a visualization technique. To evaluate the explanation map, the generated heat map was modified so that the top 5, 10, 20, 25, and 50% of pixels were 1 and the rest 0. By multiplying the original image point by point with the adjusted localization heat map, a visual explanation map was created. Figure 6 displays the visual explanation map generated by our method, which modified 25% of the original image's pixels. We examined the effectiveness of heatmaps created by XAI method using the top x percent pixels, rather than the visual explanation maps of other XAI methods. This guaranteed that one technique would outperform another not only in terms of highlighting more pixels but also in terms of capturing more relevant information for the same number of pixels.

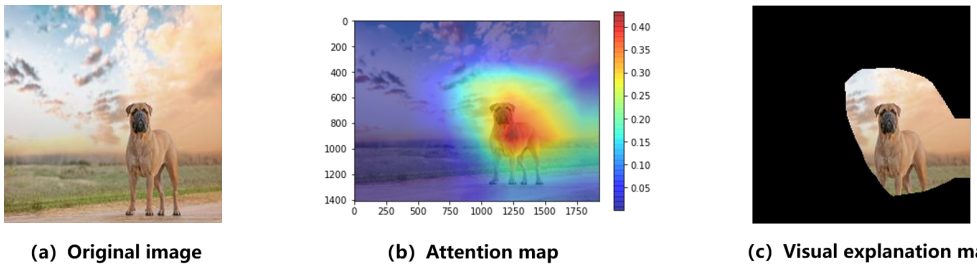

Figure 6. (**a**) Original image of a bull mastiff. (**b**) The heat map generated by the proposed method. (**c**) The attention-map-guided visual explanation map.

We evaluated the performances of explanation maps produced by our method and other XAI methods using three metrics: (a) Average drop in activation score. (b) Percent increase in activation score. (c) Percentage in metric. All the results were computed on the ImageNet dataset using Resnet-50 models.

4.2.1. Average Drop in Activation Score

An excellent explanation map will cover the majority of the elements of the object in the image that are important for making a choice. As a result, a better explanation map, rather than a whole image, should result in a small decline in the model's output scores. In Equation (8), the metric is given as the percentage drop in the model's score when only an explanation map is provided as input.

$$\text{Average drop} = \frac{1}{N}\sum_{i=1}^{N}\frac{max\left(0, Y_i^c - O_i^c\right)}{Y_i^c} * 100 \tag{8}$$

where Y_i^c is the activation score when original image i is provided as input and O_i^c is the activation score when explanation map is provided as input. N is the total number of images in the data.

4.2.2. Percent Increase in Activation Score

When the context acts as noise for the class, it has been discovered that presenting the explanation map instead of the whole image boosts the output activation scores. When only an explanation map is provided as input for a whole dataset, this measure is defined as the rate at which the model's output score rises. Formally, this can be expressed as:

$$\text{Rate of increase in scores} = \sum_{i=1}^{N} \left(\frac{1Y_i^c < O_i^c}{N} \right) * 100 \tag{9}$$

where $1Y_i^c < O_i^c$ is an indicator function that returns 1 when an argument is true. Table 3 indicates that our method has a lower average drop and higher percent increase.

Table 3. Comparison between average drop and percent increase.

Metric	Grad-CAM	Grad-CAM++	SmoothGrad-CAM++	XGrad-CAM	Ours
Average drop	45.27%	44.35%	44.82%	46.33%	42.52%
Percent increase	23.06%	23.15%	23.75%	22.15%	25.35%

4.2.3. Percentage in Metric

We created a new metric to demonstrate the results of our experiments. One of the key reasons we created this metric is because it allows a more intuitive view of how well the XAI method performs. The percentage specifies how much to mask the input image, and this image is fed into the original Resnet-50 model to check the performance of the XAI method. Using this metric, it is possible to visualize how well the XAI method performs and provide a visualization of the results from the user's perspective. Table 4 shows the results of our experiment.

Table 4. The results of using the percentage in metric.

Metric	Grad-CAM	Grad-CAM++	SmoothGrad-CAM++	XGrad-CAM	Ours
5%	0.2846	0.2715	0.4997	0.2915	1.0229
10%	0.3954	0.3841	0.7814	0.4521	1.2898
20%	4.5524	4.5001	4.8335	4.7527	5.5308
25%	6.6328	7.1551	6.7879	7.6782	9.5161
50%	8.9952	8.9876	8.7792	8.6304	11.5334

5. Results

As shown in Figure 7, the proposed method gave the clearest explanation of particular features the model learned. For instance, proposed method was able to find the most important portion of the bull mastiff's head. Additionally, the proposed method captured a larger amount of the class object (as seen in the dog image in Figure 7) and performed localization well. Table 3 shows that if we use average drop and percent increase to evaluate our method, it is better than the other XAI methods. A good explanation map will focus on most of the relevant parts of the object in the image. As a result, when we input an explanation map to the DNN model, it is expected to result in a low average drop and high percent increase. The full explanation map is used as the input, and the Resnet-50 model will provide a class score. If the explanation map concentrates on the most essential area in the image, the Resnet-50 model will provide a high-class score. According to Equations (8) and (9), as the explanation map performs better, the average drop will be lower and the percent increase will be higher.

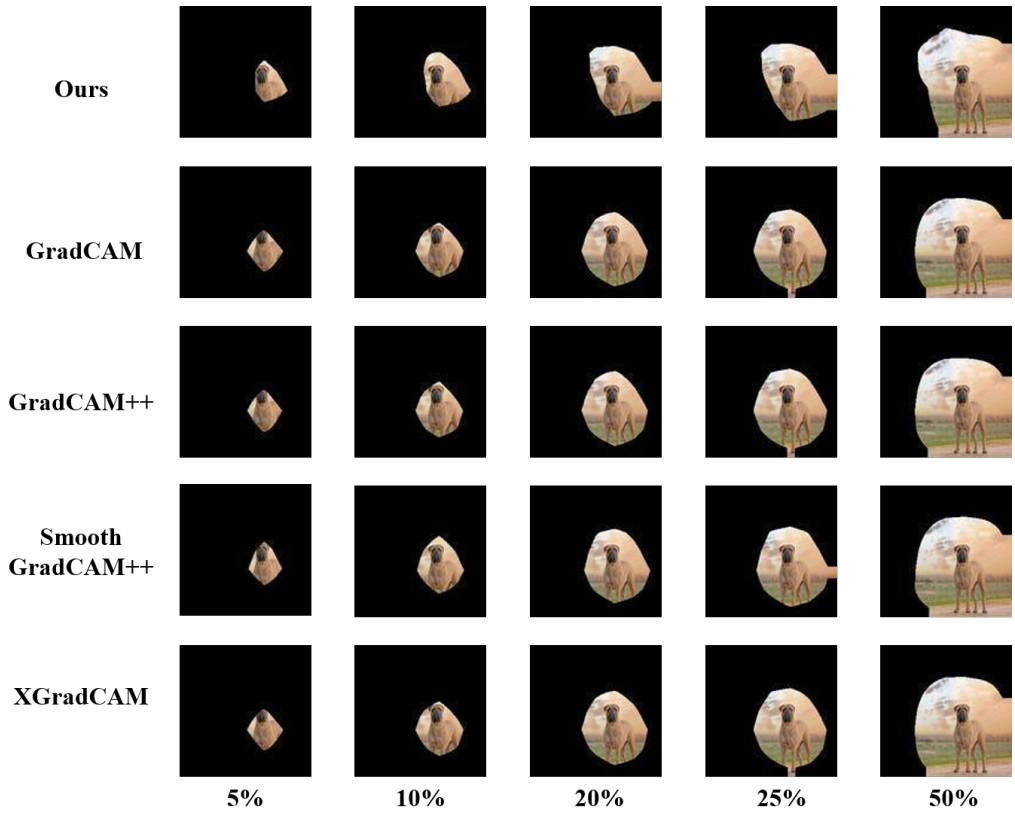

Figure 7. The results of the percentage in metric with 5%, 10%, 20%, 25%, and 50%.

6. Discussion

The computing time needed to create a single attention-map-guided visual explanation map is longer than that required by other XAI methods. The reason for this is that we employ the attention mechanism to get a higher-level feature region for each feature map. Second, as seen in Tables 3 and 4, when we try to explain models such as Resnet-50, which do not have any fully-connected layers, our methods perform only slightly better than other XAI methods. As shown in Figure 7, our method focused on more of the important region than the other XAI methods. The bull-mastiff's head was totally obtained in the five-percent and ten-percent images. This indicates that for the Resnet-50 model, the features expressed in the head region of the bull-mastiff are most important.

7. Conclusions

In this work, we proposed a novel technique—attention-map-guided visual explanation—to produce explanation maps to explain the individual decisions of CNN-based models. It uses the Grad-CAM method to extract the feature map for a deep learning model, and then uses the attention mechanism to extract the high-level attention map. We showed through objective evaluations that our method performs better than the existing state-of-the-art XAI methods. In the future, we hope to apply the proposed method to medical diagnostics, and by explaining deep learning models, we hope to persuade doctors and patients of the veracity of good deep learning models' decisions. Our study has some limitations, in light of which our findings need to be interpreted carefully. First, as in most empirical studies, the research presented here was limited by the black-box used. Second, the attention mechanism highlights some image regions which are true influences, but some are spurious.

Author Contributions: Conceptualization, J.A. and I.J.; methodology, J.A.; software, J.A.; validation, J.A. and I.J.; investigation, J.A.; resources, J.A.; data curation, J.A.; writing—original draft preparation, J.A.; writing—review and editing, J.A. and I.J.; visualization, J.A.; supervision, I.J.; project administration, I.J.; funding acquisition, I.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (NO. 2020-0-00107, Development of the technology to automate the recommendations for big data analytic models that define data characteristics and problems).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
2. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 10012–10022.
3. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 91–99. [CrossRef] [PubMed]
4. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
5. Beery, S.; Wu, G.; Rathod, V.; Votel, R.; Huang, J. Context R-CNN: Long Term Temporal Context for Per-Camera Object Detection Supplementary Material. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020.
6. Liu, C.; Chen, L.C.; Schroff, F.; Adam, H.; Hua, W.; Yuille, A.L.; Fei-Fei, L. Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019; pp. 82–92.
7. Zhang, P.; Li, X.; Hu, X.; Yang, J.; Zhang, L.; Wang, L.; Choi, Y.; Gao, J. Vinvl: Revisiting visual representations in vision-language models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 5579–5588.
8. Prakash, A.; Chitta, K.; Geiger, A. Multi-Modal Fusion Transformer for End-to-End Autonomous Driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 7077–7087.
9. Zhang, Q.; Rao, L.; Yang, Y. Group-CAM: Group Score-Weighted Visual Explanations for Deep Convolutional Networks. *arXiv* **2021**, arXiv:2103.13859.
10. Petsiuk, V.; Das, A.; Saenko, K. RISE: Randomized Input Sampling for Explanation of Black-box Models. *arXiv* **2018**, arXiv:1806.07421.
11. Ribeiro, M.T.; Singh, S.; Guestrin, C. "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, 13–17 August 2016; pp. 1135–1144.
12. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 618–626.
13. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning deep features for discriminative localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2921–2929.
14. Zhang, N.; Donahue, J.; Girshick, R.; Darrell, T. Part-based R-CNNs for fine-grained category detection. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 834–849.
15. Smilkov, D.; Thorat, N.; Kim, B.; Viégas, F.; Wattenberg, M. Smoothgrad: Removing noise by adding noise. *arXiv* **2017**, arXiv:1706.03825.
16. Omeiza, D.; Speakman, S.; Cintas, C.; Weldemariam, K. Smooth Grad-CAM++: An Enhanced Inference Level Visualization Technique for Deep Convolutional Neural Network Models. *arXiv* **2019**, arXiv:1908.01224.
17. Fu, R.; Hu, Q.; Dong, X.; Guo, Y.; Gao, Y.; Li, B. Axiom-based Grad-CAM: Towards Accurate Visualization and Explanation of CNNs. *arXiv* **2020**, arXiv:2008.02312.
18. Luong, M.T.; Pham, H.; Manning, C.D. Effective Approaches to Attention-based Neural Machine Translation. *arXiv* **2015**, arXiv:1508.04025.

19. Liu, G.; Guo, J. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing* **2019**, *337*, 325–338. [CrossRef]
20. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
21. Wang, F.; Jiang, M.; Qian, C.; Yang, S.; Li, C.; Zhang, H.; Wang, X.; Tang, X. Residual attention network for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3156–3164.
22. Linsley, D.; Shiebler, D.; Eberhardt, S.; Serre, T. Global-and-local attention networks for visual recognition. *Benefits* **2018**, *64*, 1.
23. Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*; PMLR: New York City, NY, USA, 2015; pp. 2048–2057.
24. Yang, Z.; Li, Y.; Yang, J.; Luo, J. Action recognition with spatio–temporal visual attention on skeleton image sequences. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *29*, 2405–2415. [CrossRef]
25. Kim, J.; Canny, J. Interpretable learning for self-driving cars by visualizing causal attention. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2942–2950.
26. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
27. Chen, L.; Zhang, H.; Xiao, J.; Nie, L.; Shao, J.; Liu, W.; Chua, T.S. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5659–5667.
28. Zagoruyko, S.; Komodakis, N. Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer. *arXiv* **2016**, arXiv:1612.03928.
29. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
30. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
31. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
32. Desai, S.; Ramaswamy, H.G. Ablation-CAM: Visual Explanations for Deep Convolutional Network via Gradient-free Localization. In Proceedings of the 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), Snowmass Village, CO, USA, 1–5 March 2020; pp. 972–980.