Axiom-based Grad-CAM: Towards Accurate Visualization and Explanation of CNNs Appendix

1 Proof

In this section, we aim to demonstrate that given an arbitrary layer in ReLU-CNNs, for any class of interest, there exists a specific equation between the class score and the feature maps of the layer.

For a ReLU-CNN which only has ReLU rectification as its nonlinearity, the following equation holds for an arbitrary layer l:

$$u_j^{l+1} = \sum_i \left(\frac{\partial u_j^{l+1}}{\partial u_i^l} u_i^l \right) + b_j^{l+1}, \tag{1}$$

where u_i^l represents an unit in layer l, u_j^{l+1} represents an unit in layer $l+1$, $\frac{\partial u_j^{l+1}}{\partial u_i^l}$ is the gradient of u_j^{l+1} w.r.t. u_i^l, b_j^{l+1} is the bias term asscociated with the unit u_j^{l+1}. Note that, if unit u_j^t is an output of a ReLU or pooling layer, the corresponding bias term b_j^t is zero.

We then prove our statement (i.e., Eq.(5) in the main paper) using *mathematical induction* [1]. In the top layer L, the response of the c-th unit is exactly the class score of interest S_c in the main paper, and it is easy to verify that:

$$u_c^L = \sum_i \left(\frac{\partial u_c^L}{\partial u_i^{L-1}} u_i^{L-1} \right) + b_c^L, \tag{2}$$

Suppose that for layer l ($l < L$):

$$u_c^L = \sum_i \left(\frac{\partial u_c^L}{\partial u_i^l} u_i^l \right) + \sum_{t=l+1}^{L} \sum_k \frac{\partial u_c^L}{\partial u_k^t} b_k^t, \tag{3}$$

Then, for layer $l-1$, it holds:

$$
\begin{aligned}
&\sum_{i'}\left(\frac{\partial u_c^L}{\partial u_{i'}^{l-1}}u_{i'}^{l-1}\right)+\sum_{t=l}^{L}\sum_{k'}\frac{\partial u_c^L}{\partial u_{k'}^{t}}b_{k'}^{t}\\
&=\sum_{i'}\left(\frac{\partial u_c^L}{\partial u_{i'}^{l-1}}u_{i'}^{l-1}\right)+\sum_{k'}\frac{\partial u_c^L}{\partial u_{k'}^{l}}b_{k'}^{l}+\sum_{t=l+1}^{L}\sum_{k}\frac{\partial u_c^L}{\partial u_{k}^{t}}b_{k}^{t}\\
&=\sum_{i'}\left(\sum_{i}\left(\frac{\partial u_c^L}{\partial u_i^l}\frac{\partial u_i^l}{\partial u_{i'}^{l-1}}\right)u_{i'}^{l-1}\right)+\sum_{k'}\frac{\partial u_c^L}{\partial u_{k'}^{l}}b_{k'}^{l}+\sum_{t=l+1}^{L}\sum_{k}\frac{\partial u_c^L}{\partial u_{k}^{t}}b_{k}^{t}\\
&=\sum_{i}\frac{\partial u_c^L}{\partial u_i^l}\left(\sum_{i'}\left(\frac{\partial u_i^l}{\partial u_{i'}^{l-1}}u_{i'}^{l-1}\right)\right)+\sum_{k'}\frac{\partial u_c^L}{\partial u_{k'}^{l}}b_{k'}^{l}+\sum_{t=l+1}^{L}\sum_{k}\frac{\partial u_c^L}{\partial u_{k}^{t}}b_{k}^{t}\\
&=\sum_{i}\frac{\partial u_c^L}{\partial u_i^l}\left(\sum_{i'}\left(\frac{\partial u_i^l}{\partial u_{i'}^{l-1}}u_{i'}^{l-1}\right)+b_i^l\right)+\sum_{t=l+1}^{L}\sum_{k}\frac{\partial u_c^L}{\partial u_{k}^{t}}b_{k}^{t}\\
&=\sum_{i}\left(\frac{\partial u_c^L}{\partial u_i^l}u_i^l\right)+\sum_{t=l+1}^{L}\sum_{k}\frac{\partial u_c^L}{\partial u_{k}^{t}}b_{k}^{t}
\end{aligned}
\tag{4}
$$

i.e.,

$$
u_c^L=\sum_{i'}\left(\frac{\partial u_c^L}{\partial u_{i'}^{l-1}}u_{i'}^{l-1}\right)+\sum_{t=l}^{L}\sum_{k'}\frac{\partial u_c^L}{\partial u_{k'}^{t}}b_{k'}^{t}, \tag{5}
$$

This means that for an arbitrary layer, the class score equals to the sum of gradient$\times$feature plus an extra bias term.

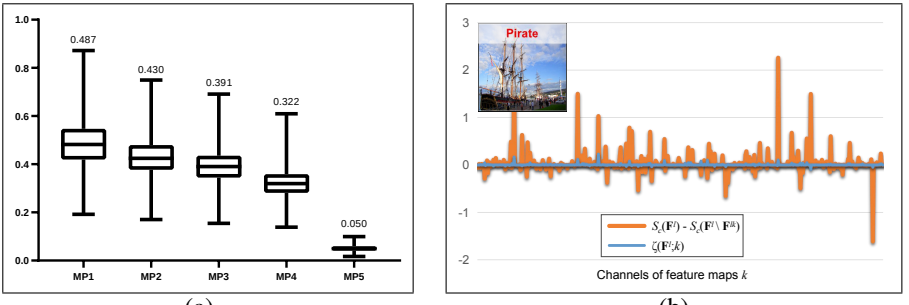
2 $\varepsilon(\mathbf{F}^l)$ and $\zeta(\mathbf{F}^l;k)$

$\varepsilon(\mathbf{F}^l)$ is the bias term in Eq. (5) in the main paper. We calculated $\left|\frac{\varepsilon(\mathbf{F}^l)}{S_c(\mathbf{F}^l)}\right|$ of 1000 input images in different layers of VGG-16 model, with the class of interest c set as the top-1 predicted class. Fig. 1(a) shows that this term is rather large in shallow layers.

$\zeta(\mathbf{F}^l;k)$ is a bias term in Eq. (6) in the main paper. Given an input example, Fig. 1(b) shows the values of $S_c(\mathbf{F}^l)-S_c(\mathbf{F}^l\backslash\mathbf{F}^{lk})$ and $\zeta(\mathbf{F}^l;k)$ w.r.t. all the feature maps in the last spatial layer of VGG16 model. It can be seen that $\frac{|\zeta(\mathbf{F}^l;k)|}{|S_c(\mathbf{F}^l)-S_c(\mathbf{F}^l\backslash\mathbf{F}^{lk})|}$ is rather small for most of the feature maps. Exceptions usually happen in the unimportant feature maps whose removing only lead to a tiny score change.

3 CAM, Grad-CAM, Ablation-CAM and XGrad-CAM on GAP-CNNs

In this section, we prove that for GAP-CNNs (e.g., ResNet-101, Inception_v3), CAM [5], Grad-CAM [3], Ablation-CAM [1] and our XGrad-CAM achieve the same performance on the last spatial layers of the models.



(a) (b)

Figure 1: (a) Normalized $\varepsilon(\mathbf{F}^l)$ in different layers of VGG-16 model. "MP" represents for Maxpooling layer. The mean values are provided above the box-plots; (b) $\frac{|\zeta(\mathbf{F}^l;k)|}{|S_c(\mathbf{F}^l)-S_c(\mathbf{F}^l\setminus\mathbf{F}^{lk})|}$ is small for most of the feature maps. Exceptions usually happen in the unimportant feature maps.

GAP-CNNs usually consist of fully-convolution layers, global average pooling and a linear classifier with softmax. Specifically, let $\mathbf{F}^l$ be the last spatial layer, the output of the global average pooling is:

$$A^k = \frac{1}{Z}\sum_{x,y} F^{lk}(x,y) \tag{6}$$

where Z is the number of units in the k-th feature map. The score of class c is exactly a weighted sum of A^k since the classifier is linear:

$$S_c = \sum_{k=1}^{K}\left(w_c^k A^k\right) + b_c \tag{7}$$

where w_c^k is the weight connecting the k-th feature map with the c-th class, b_c is a bias. Combining Eq. (6) and Eq. (7), we have:

$$S_c(\mathbf{F}^l) = \frac{1}{Z}\sum_{x,y}\sum_{k=1}^{K}\left(w_c^k F^{lk}(x,y)\right) + b_c \tag{8}$$

The weight of CAM [5] is then defined as w_c^k.

For a GAP-CNN, we can simply get that $\forall x,y$, $\frac{\partial S_c(\mathbf{F}^l)}{\partial F^{lk}(x,y)} = \frac{1}{Z}w_c^k$ using the Chain Rule. Recall the definition of the weights in Grad-CAM [3], we have:

$$\frac{1}{Z}\sum_{x,y}\frac{\partial S_c(\mathbf{F}^l)}{\partial F^{lk}(x,y)} = \frac{1}{Z}w_c^k. \tag{9}$$

Recall the definition of the weights in Ablation-CAM [1], we have:

$$\frac{S_c(\mathbf{F}^l) - S_c(\mathbf{F}^l\setminus\mathbf{F}^{lk})}{\sum_{x,y}F^{lk}(x,y)} = \frac{w_c^k A^k}{ZA^k} = \frac{1}{Z}w_c^k. \tag{10}$$

Recall the definition of the weights in XGrad-CAM, we have:

$$\sum_{x,y}\left(\frac{F^{lk}(x,y)}{\sum_{x,y}F^{lk}(x,y)}\frac{\partial S_c(\mathbf{F}^l)}{\partial F^{lk}(x,y)}\right) = \sum_{x,y}\left(\frac{F^{lk}(x,y)}{\sum_{x,y}F^{lk}(x,y)}\frac{1}{Z}w_c^k\right) = \frac{1}{Z}w_c^k. \tag{11}$$

It shows that the weights of Grad-CAM [3], Ablation-CAM [1] and XGrad-CAM are exactly the same in the case of GAP-CNNs. Besides, they are also identical to the weight of CAM [5] expect a constant Z, which makes no difference for visualization. Therefore, we can conclude that CAM [5], Grad-CAM [3], Ablation-CAM [1] and XGrad-CAM achieve the same performance on the last spatial layers of GAP-CNNs.

4 Additional Visualization Results

In the section of class discrimination analysis in the main paper, we evaluated the class-discriminability of different CAM methods using their guided versions rather than themselves. The motivation comes from two aspects. First, the guided versions have the same class-discriminability as the original versions. As shown in Fig. 2, we visualized several visualization results of XGrad-CAM, Guided Backprop [4] and Guided XGrad-CAM. It is shown that Guided XGrad-CAM inherits the class-discriminability of XGrad-CAM completely. This phenomenon applies to all the other CAM methods. Second, the results of guided versions provides a better visualization for the objects of interest with more object details. It helps the subjects make their decisions more accurately and efficiently in the game of "What do you see" as shown in Fig.4(a) in the main paper.

Fig. 3 presents several qualitative results in VOC 2007 validation set to further compare the class-discriminability of different CAM methods. We can see that if there are objects belonging to multiple classes in an image, Grad-CAM++ also highlights regions of irrelevant classes. Clearly, Grad-CAM++ is not class-discriminative compared with the other three CAM methods.

References

[1] Saurabh Desai and Harish G Ramaswamy. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. In *WACV*, pages 972–980, 2020.

[2] Pieter-Jan Kindermans, Kristof Schütt, Klaus-Robert Müller, and Sven Dähne. Investigating the influence of noise and distractors on the interpretation of neural networks. *arXiv preprint arXiv:1611.07270*, 2016.

[3] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *ICCV*, pages 618–626, 2017.

[4] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[5] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, pages 2921–2929, 2016.

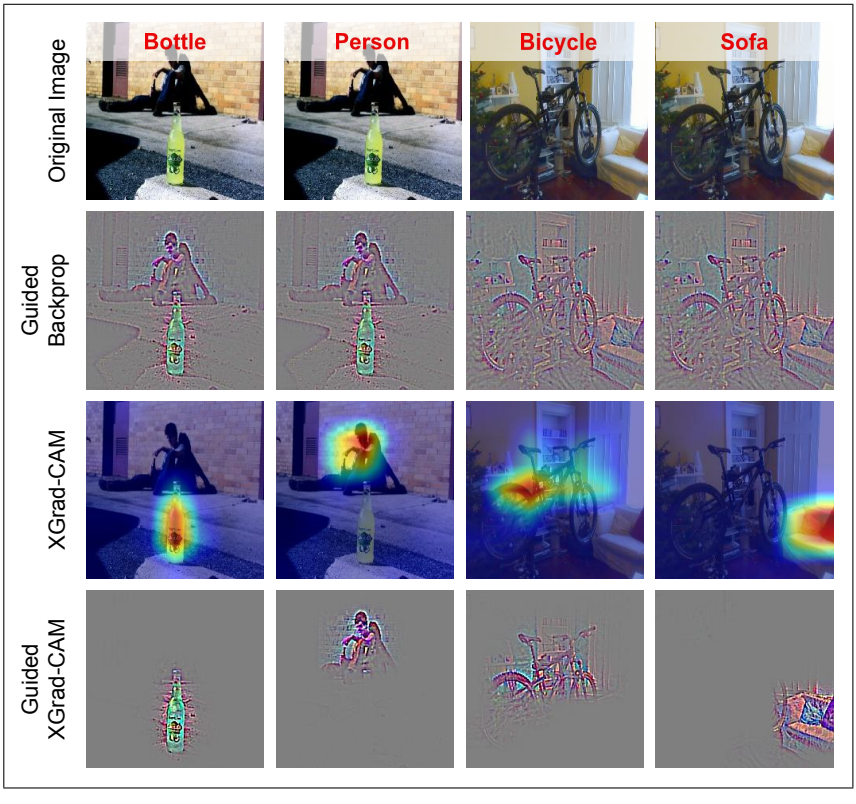


Figure 2: Several visualization results of XGrad-CAM, Guided Backprop and Guided XGrad-CAM.

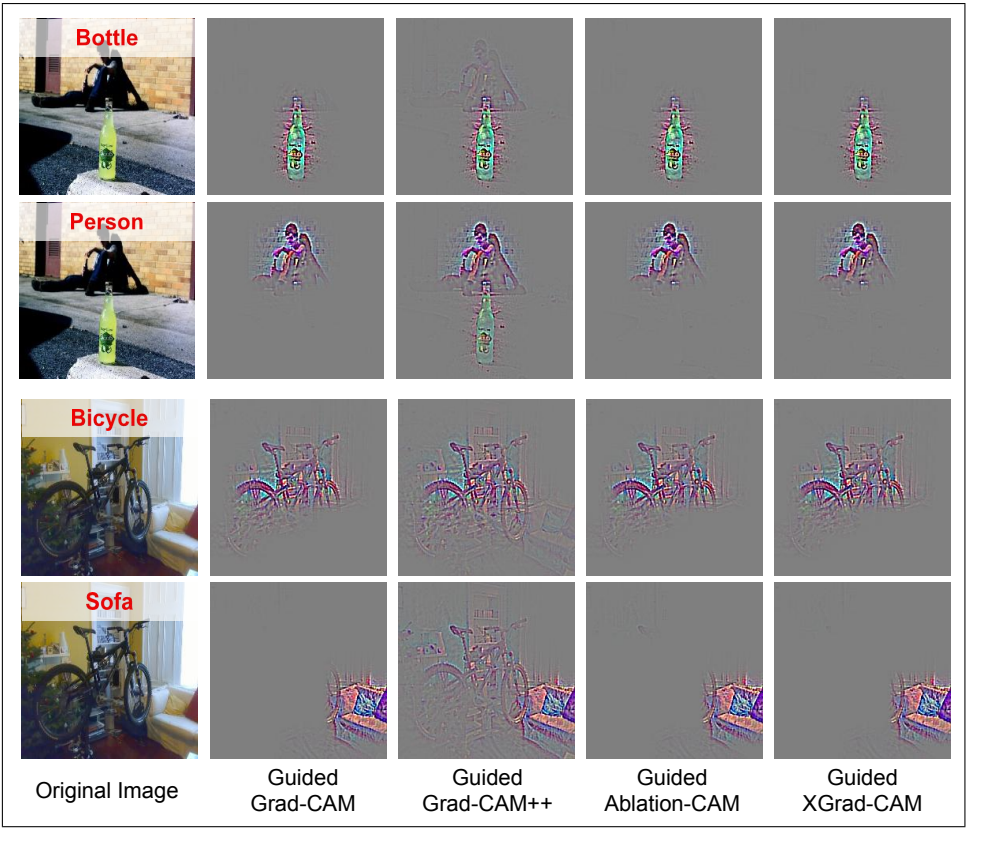


Figure 3: Additional visualization results to compare the class-discriminability of different CAM methods.