

Grad-CAM: Why did you say that?

Ramprasaath R. Selvaraju

Abhishek Das

Devi Parikh

Ramakrishna Vedantam

Michael Cogswell

Virginia Tech

{ram21, abhshkdz, vrama91, cogswell, parikh, dbatra}@vt.edu

Abstract

We propose a technique for making Convolutional Neural Network (CNN)-based models more transparent by visualizing input regions that are ‘important’ for predictions – producing visual explanations. Our approach, called Gradient-weighted Class Activation Mapping (Grad-CAM), uses class-specific gradient information to localize important regions. These localizations are combined with existing pixel-space visualizations to create a novel high-resolution and class-discriminative visualization called Guided Grad-CAM. These methods help better understand CNN-based models, including image captioning and visual question answering (VQA) models. We evaluate our visual explanations by measuring their ability to discriminate between classes, to inspire trust in humans, and their correlation with occlusion maps. Grad-CAM provides a new way to understand CNN-based models.

We have released code, an online demo hosted on CloudCV [1], and the full paper [8].¹

1. Introduction

Convolutional Neural Networks (CNNs) and other deep networks have enabled unprecedented breakthroughs in a variety of computer vision tasks, from image classification to image captioning and visual question answering. While these deep neural networks enable superior performance, their lack of decomposability into *intuitive and understandable* components makes them hard to interpret. Consequently, when today’s intelligent systems fail, they fail spectacularly disgracefully, without warning or explanation, leaving a user staring at incoherent output, wondering why the system did what it did. In order to build trust in intelligent systems and move towards their meaningful integration into our everyday lives, it is clear that we must build ‘transparent’ models that explain why they predict what they do.

However, there is a trade-off between accuracy and simplicity/interpretability. Classical rule-based or expert sys-

tems were highly interpretable but not very accurate (or robust). Decomposable pipelines where each stage is hand-designed are thought to be more interpretable as each individual component assumes a natural intuitive explanation. This tradeoff is realized in more recent work, like Class Activation Mapping (CAM) [12], which allows explanations for a specific class of image classification CNNs. By using deep models, we sacrifice a degree of interpretability in pipeline modules in order to achieve greater performance through greater abstraction (more layers) and tighter integration (end-to-end training).

What makes a good visual explanation? Consider image classification – a ‘good’ visual explanation from the model justifying a predicted class should be (a) class-discriminative (*i.e.* localize the category in the image) and (b) high-resolution (*i.e.* capture fine-grained detail). To be concrete we introduce Grad-CAM using the notion ‘class’ from image classification (*e.g.*, cat or dog), but visual explanations can be considered for any differentiable node in a computational graph, including words from a caption or the answer to a question.

This is illustrated in Fig. 1, where we visualize the ‘tiger cat’ class. Pixel-space gradient visualizations such as Guided Backpropagation [9], seen at the top of Fig. 1, are high-resolution and highlight fine-grained details in the image, but are not class-discriminative. Both the cat and the dog are highlighted despite ‘tiger cat’ being the class of interest (in fact, the Guided Backpropagation visualizations of ‘boxer’ dog ‘tiger cat’ are indistinguishable). However, the Grad-CAM visualization is low-resolution and does not contain fine details. A high-res visualization like Guided Grad-CAM helps show these details by highlighting stripes in the cat in addition to localizing the cat. We combine the best of both worlds by fusing existing pixel-space gradient visualizations with our novel localization method – called Grad-CAM – to create Guided Grad-CAM visualizations, which are both high-resolution and class-discriminative.

In this abstract: (1) We propose Gradient-weighted Class Activation Mapping (Grad-CAM) to generate visual explanations from *any* CNN-based network without requiring ar-

¹ code: <https://github.com/ramptrs/grad-cam/> demo: <http://gradcam.cloudcv.org> paper: <https://arxiv.org/abs/1610.02391>

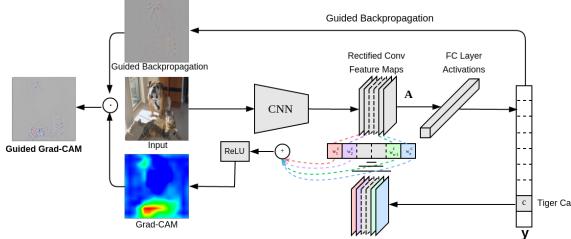


Figure 1: Grad-CAM overview: Given an image, and a category (‘tiger cat’) as input, we forward propagate the image through the model to obtain the raw class scores before softmax. The gradients are set to zero for all classes except the desired class (tiger cat), which is set to 1. This signal is then backpropagated to the rectified convolutional feature map of interest, where we can compute the coarse Grad-CAM localization (blue heatmap). Finally, we pointwise multiply the heatmap with guided backpropagation to get Guided Grad-CAM visualizations which are both high-resolution and class-discriminative.

chitectural changes. (2) To illustrate the broad applicability of our technique across tasks, we apply Grad-CAM to state-of-the-art image captioning and visual question answering models. (3) We design and conduct human studies to show that Guided Grad-CAM explanations are class-discriminative and help humans not only establish trust, but also help untrained users successfully discern a ‘stronger’ deep network from a ‘weaker’ one *even when both networks make identical predictions, simply on the basis of their visual explanations*. (4) Our code and demos of Grad-CAM are released to help others apply Grad-CAM to interpret their own models.

2. Approach

We review Class Activation Mapping (CAM) [12], propose Grad-CAM, and combine Grad-CAM with high-resolution visualizations to form Guided Grad-CAM. This is summarized by Fig. 1.

Class Activation Mapping (CAM). CAM [12] produces a localization map from image classification CNNs where global-average-pooled convolutional feature maps are fed directly into a softmax. Specifically, let the penultimate CNN layer produce K feature maps $A^k \in \mathbb{R}^{u \times v}$ of width u and height v . These feature maps are then spatially pooled using Global Average Pooling (GAP) and linearly transformed to produce a score y^c for each class c

$$y_c = \sum_k w_k^c \frac{1}{Z} \sum_i \sum_j A_{ij}^k. \quad (1)$$

To produce a localization map $L_{\text{CAM}}^c \in \mathbb{R}^{u \times v}$ for class c , CAM computes the linear combination of the final feature maps using the learned weights of the final layer: $L_{\text{CAM}}^c = \sum_k w_k^c A^k$. This is normalized to lie between 0 and 1 for visualization purposes. CAM can not be applied to networks which use multiple fully-connected layers before the output layer, so fully-connected layers are replaced with convolutional ones and the network is re-trained. In comparison, our approach can be applied directly to any CNN-based differentiable architecture as is without re-training.

Gradient-weighted Class Activation Mapping. In order

to obtain the class-discriminative localization map Grad-CAM $L_{\text{Grad-CAM}}^c \in \mathbb{R}^{u \times v}$ in generic CNN-based architectures, we first compute the gradient of y^c with respect to feature maps A of a convolutional layer, *i.e.* $\frac{\partial y^c}{\partial A_{ij}}$. These gradients are global-average-pooled to obtain weights α_k^c :

$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}} \quad (2)$$

This weight α_k^c represents a *partial linearization* of the deep network downstream from A , and captures the ‘importance’ of feature map k for a target class c . In general, y^c need not be a class score, but could be any differentiable activation. As in CAM, our Grad-CAM heat-map is a weighted combination of feature maps, but we follow this by a ReLU:

$$L_{\text{Grad-CAM}}^c = \text{ReLU} \left(\sum_k \alpha_k^c A^k \right) \quad (3)$$

This results in a coarse heat-map, which is normalized for visualization. Other than the ReLU in (3), *Grad-CAM is a generalization of CAM* (w_k^c are precisely α_k^c where CAM can be applied) to any CNN-based architecture (CNNs with fully-connected-layers, ResNets, CNNs stacked with Recurrent Neural Networks (RNNs) *etc.*).

Guided Grad-CAM. In order to combine the class-discriminative nature of Grad-CAM and the high-resolution nature of Guided Backpropagation, we fuse them via pointwise multiplication to create Guided Grad-CAM, shown on the left of Fig. 1. We expect the last convolutional layers to have the best compromise between high-level semantics and detailed spatial information, so we use these feature maps to compute Grad-CAM and Guided Grad-CAM.

3. Experiments

We evaluate our visualization then show image captioning and visual question answering examples.

3.1. Evaluating Visualizations

Evaluating Class Discrimination. Intuitively, a good prediction explanation is one that produces discriminative visualizations for the class of interest. We select images from PASCAL VOC 2007 val set that contain exactly two annotated categories, and create visualizations for one of the classes. These are shown to workers on Amazon Mechanical Turk (AMT), who are asked “Which of the two object categories is depicted in the image?” and presented with the two categories present in the original image as options. As shown in Table. 1 column 1, human subjects can correctly identify the category being visualized substantially more often when using Grad-CAM. This makes Guided Backpropagation more class-discriminative.

Evaluating Trust. Given explanations from two different models, we want to evaluate which of them seems more trustworthy. We use AlexNet and VGG-16 to compare

Guided Backpropagation and Guided Grad-CAM visualizations, noting that VGG-16 is known to be more reliable than AlexNet. In order to tease apart the efficacy of the visualization from the accuracy of the model being visualized, we consider only those instances where *both* models made the same prediction as ground truth. Given a visualization from AlexNet, one from VGG-16, and the name of the object predicted by both the networks, workers are instructed to rate which model is more reliable. Results are shown in the Relative Reliability column of Table. 1, where scores range from -2 to +2 and positive scores indicate VGG is judged to be more reliable than AlexNet. With Guided Backpropagation, humans score VGG as slightly more reliable than AlexNet, while with Guided Grad-CAM they score VGG as clearly more reliable than AlexNet. Thus our Guided Grad-CAM visualization can help users place trust in a model that can generalize better, based on individual prediction explanations.

Faithfulness vs. Interpretability. Faithfulness of a visualization to a model is its ability to accurately explain the function learned by the model. Naturally, there exists a tradeoff between the interpretability and faithfulness for complex models operating on highly compositional inputs. A more faithful visualization might describe the model in precise detail yet be completely opaque to human inspection. Here we are only interested in local fidelity; the visualization need only explain the parts of the model relevant to that image. That is, in the vicinity of the input data point, our explanation should be faithful to the model [7].

For comparison, we need a reference explanation with high local-faithfulness. One obvious choice for such a visualization is image occlusion [11], where we measure the difference in CNN scores when patches of the input image are masked out. Interestingly, patches which change the CNN score are also patches to which Guided Grad-CAM assigns high intensity, as shown by measuring rank correlation between patch intensities in Table. 1 (3rd column). This shows that Guided Grad-CAM is more faithful to the original model than Guided Backpropagation.

Method	Human Classification Accuracy	Relative Reliability	Rank Correlation w/ Occlusion
Guided Backpropagation	44.44	+1.00	0.168
Guided Grad-CAM	61.23	+1.27	0.261

Table 1: Quantitative Visualization Evaluation. Guided Grad-CAM enables humans to differentiate between visualizations of different classes (Human Classification Accuracy) and pick more reliable models (Relative Reliability). It also accurately reflects the behavior of the model (Rank Correlation w/ Occlusion).

3.2. Analyzing Failure Modes for VGG-16

We use Guided Grad-CAM to analyze failure modes of the VGG-16 CNN on ImageNet classification [2].

In order to see what mistakes a network is making we first get a list of examples that the network (VGG-16) fails to classify correctly. For the misclassified examples, we use Guided Grad-CAM to visualize both the cor-

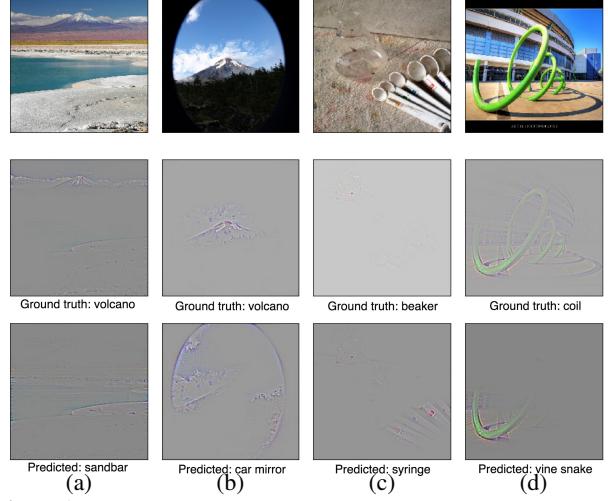


Figure 2: In these cases the model (VGG-16) failed to predict the correct class as its top 1 prediction, but it even failed to predict the correct class in its top 5 for figure b. All of these errors are due in part to class ambiguity. (a) For example, the network predicts ‘sandbar’ based on the foreground of (a), but it also knows where the correct label ‘volcano’ is located. (c-d) In other cases, these errors are still reasonable but not immediately apparent. For example, humans would find it hard to explain the predicted ‘syringes’ in (c) without looking at the visualization for the predicted class. rect class and the predicted class. A major advantage of Guided Grad-CAM over other methods is its ability to more usefully investigate and explain classification mistakes, since our visualizations are high-resolution and more class-discriminative. As seen in Fig. 2, some failures are due to ambiguities inherent in ImageNet classification. We can also see that seemingly unreasonable predictions have reasonable explanations, which is a similar observation to HOGgles [10].

3.3. Image Captioning and VQA

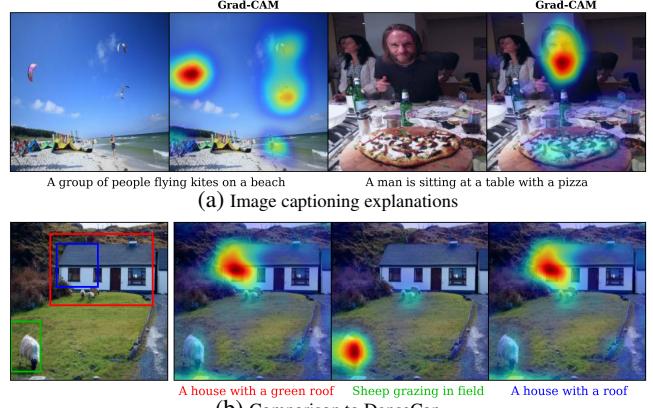


Figure 3: Interpreting image captioning models: We use our class-discriminative localization technique, Grad-CAM to find spatial support regions for captions in images. Fig. 3a Visual explanations from image captioning model [4] highlighting image regions considered to be important for producing the captions. Fig. 3b Grad-CAM localizations of a *global* or *holistic* captioning model for captions generated by a dense captioning model [3] for the three bounding box proposals marked on the left. We can see that we get back Grad-CAM localizations (right) that agree with those bounding boxes – even though the captioning model and Grad-CAM do not use any bounding box annotations.

Image Captioning. We visualize spatial support for a simple image captioning model [4]² (without attention) using

²‘neuraltalk2’, publicly available at <https://github.com/>

Grad-CAM visualizations. Given a caption, we compute the gradient of its log probability w.r.t. units in the last convolutional layer of the CNN (*conv5_3* for VGG-16) and generate Grad-CAM visualizations as described in Section 2. Results are shown in Fig. 3a. In the first example, the Grad-CAM maps for the generated caption localize every occurrence of both the kites and people in spite of their relatively small size. In the top right example, Grad-CAM correctly highlights the pizza and the man, but ignores the woman nearby, since ‘woman’ is not mentioned in the caption. As described in Fig. 3b, if we generate captions for specific bounding boxes in an image then Grad-CAM highlights only regions within those bounding boxes.

Visual Question Answering. Typical VQA pipelines consist of a CNN to model images and an RNN language model for questions. The image and the question representations are fused to predict the answer, typically with a 1000-way classification. Since this is a classification problem, we pick

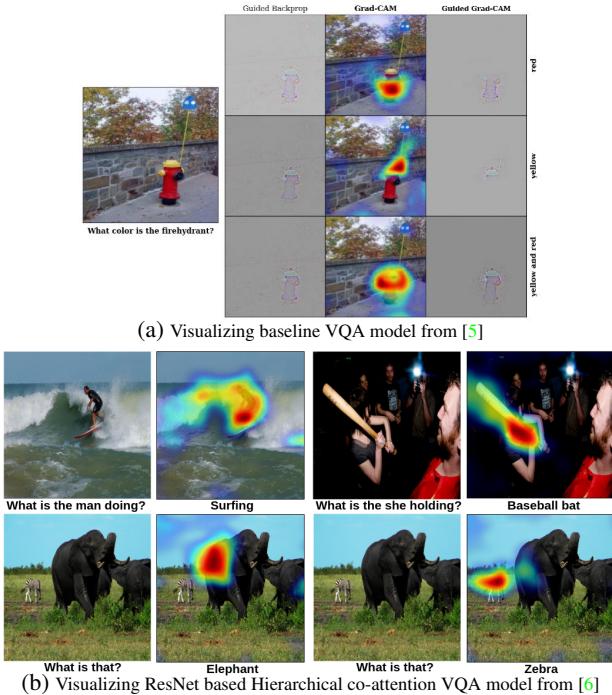


Figure 4: VQA visualizations: (a) Given the image on the left and the question “What color is the firehydrant?”, we visualize Grad-CAMs and Guided Grad-CAMs for the answers “red”, “yellow” and “yellow and red”. Our visualizations are highly interpretable and help explain the model’s predictions – for “red”, the model focuses on the bottom red part of the firehydrant; when forced to answer “yellow”, the model concentrates on its top yellow cap, and when forced to answer “yellow and red”, it looks at the whole firehydrant! (b) Grad-CAM for ResNet-based VQA model.

an answer (the score y_c in (1)) and use its score to compute Grad-CAM. Despite the complexity of the task, involving both visual and language components, the explanations (of the baseline VQA model from [5] and a ResNet based hierarchical co-attention model from [6]) described in Fig. 4 are surprisingly intuitive and informative.

4. Conclusion

In this work, we proposed a novel class-discriminative localization technique – Gradient-weighted Class Activation Mapping (Grad-CAM) – and combined it with existing high-resolution visualizations to produce visual explanations for CNN-based models. Human studies reveal that our localization-augmented visualizations can discriminate between classes more accurately and better reveal the trustworthiness of a classifier. In addition to the image captioning and VQA examples shown here, the full version of our paper [8] evaluates Grad-CAM on the ImageNet localization challenge, analyzes failure modes of VGG-16 on ImageNet classification using Grad-CAM, measures correlation between VQA Grad-CAM and human attention maps, describes ablation studies and provides many more examples. Grad-CAM provides a new way to understand any CNN-based model.

References

- [1] H. Agrawal, C. S. Mathialagan, Y. Goyal, N. Chavali, P. Banik, A. Mohapatra, A. Osman, and D. Batra. CloudCV: Large Scale Distributed Computer Vision as a Cloud Service. In *Mobile Cloud Visual Media Computing*, pages 265–290. Springer, 2015. 1
- [2] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009. 3
- [3] J. Johnson, A. Karpathy, and L. Fei-Fei. DenseCap: Fully Convolutional Localization Networks for Dense Captioning. In *CVPR*, 2016. 3
- [4] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015. 3
- [5] J. Lu, X. Lin, D. Batra, and D. Parikh. Deeper LSTM and normalized CNN Visual Question Answering model. https://github.com/VT-vision-lab/VQA_LSTM_CNN, 2015. 4
- [6] J. Lu, J. Yang, D. Batra, and D. Parikh. Hierarchical Question-Image Co-Attention for Visual Question Answering. In *NIPS*, 2016. 4
- [7] M. T. Ribeiro, S. Singh, and C. Guestrin. “Why Should I Trust You?”: Explaining the Predictions of Any Classifier. In *SIGKDD*, 2016. 3
- [8] R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra. Grad-CAM: Why did you say that? Visual Explanations from Deep Networks via Gradient-based Localization. *CoRR*, abs/1610.02391, 2016. 1, 4
- [9] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. A. Riedmiller. Striving for Simplicity: The All Convolutional Net. *CoRR*, abs/1412.6806, 2014. 1
- [10] C. Vondrick, A. Khosla, T. Malisiewicz, and A. Torralba. HOGgles: Visualizing Object Detection Features. *ICCV*, 2013. 3
- [11] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014. 3
- [12] B. Zhou, A. Khosla, L. A., A. Oliva, and A. Torralba. Learning Deep Features for Discriminative Localization. In *CVPR*, 2016. 1, 2