Explainable Models with Consistent Interpretations

Vipin Pillai, Hamed Pirsiavash
University of Maryland, Baltimore County
{vp7, hpirsiav}@umbc.edu

Abstract

Given the widespread deployment of black box deep neural networks in computer vision applications, the interpretability aspect of these black box systems has recently gained traction. Various methods have been proposed to explain the results of such deep neural networks. However, some recent works have shown that such explanation methods are biased and do not produce consistent interpretations. Hence, rather than introducing a novel explanation method, we learn models that are encouraged to be interpretable given an explanation method. We use Grad-CAM as the explanation algorithm and encourage the network to learn consistent interpretations along with maximizing the log-likelihood of the correct class. We show that our method outperforms the baseline on the pointing game evaluation on ImageNet and MS-COCO datasets respectively. We also introduce new evaluation metrics that penalize the saliency map if it lies outside the ground truth bounding box or segmentation mask, and show that our method outperforms the baseline on these metrics as well. Moreover, our model trained with interpretation consistency generalizes to other explanation algorithms on all the evaluation metrics.

1 Introduction

Deep learning has achieved great results in various applications including computer vision. In many applications, particularly in safety critical systems, it is very important to understand the underlying decision making process of the model (Alexandrov 2017; Vellido 2019). For instance, when a self-driving car makes a wrong decision resulting in an accident, we want to be able to investigate what part of the input influenced this decision. As another example, in medical applications, it is not easy to utilize the results of deep learning models without knowing the underlying reasoning process in a human interpretable way (Caruana et al. 2015). Hence, building network interpretation methods have become an active research area. Most such methods focus on using a heatmap to describe the interpretation: Given a model, an input image, and a decision, e.g., an output category in classification setting, the interpretation method generates a heatmap in the size of the input image that has larger values on the regions of the image that has influenced the model to make the given decision.

Recent works (Adebayo et al. 2018; Subramanya, Pillai, and Pirsiavash 2019) have shown that the result of most current model interpretation methods is not consistent with our prior knowledge about understanding of an image, and hence are difficult to trust. We believe this inconsistency may have at least two sources: (1) the interpretation method does not produce correct interpretation, (2) the model is not making the decision as we want (e.g., it has learned some unwanted bias from the training data). For instance, in training an image classifier for "fish" category, if a person is holding the fish in most training examples for "fish" category, then the model may decide to use features of the person's hand to detect the fish. This is an example of association bias in the training data. However, it is unwanted since we know that such biases may not exist in some final test images (Singh et al. 2020).

There is a large community developing better interpretation algorithms to solve the first point above, so in this work, we assume the interpretation method is given and try to focus on resolving the second point above. Specifically, we train deep models such that under a "given" interpretation method, the network produces a reasonable interpretation that is consistent. This means the model should only use the information that we believe is relevant to the model's final decision. We would like to emphasize that we are not improving the interpretation algorithm as many other works do. Instead, we are learning a more trustworthy model under a given interpretation method.

If we knew the ground truth interpretation for a decision, this goal can be achieved by simply supervising the interpretation output (Ross, Hughes, and Doshi-Velez 2017). However, we do not always have access to the ground truth interpretation. There are some works that assume the interpretation mask should be close to the object's segmentation mask (Li et al. 2019) and use the segmentation mask in the supervision. However, we believe that such a hard constraint might not always be necessary as the network can make decisions by relying solely on the most discriminative parts of an object instead of the whole object area. For example, the model can use only the head of a dog instead of its whole body to reason about dog classification. Moreover, such methods require the segmentation mask during training, which is generally expensive. To mitigate this lack of ground truth masks, we adopt ideas from self-supervised

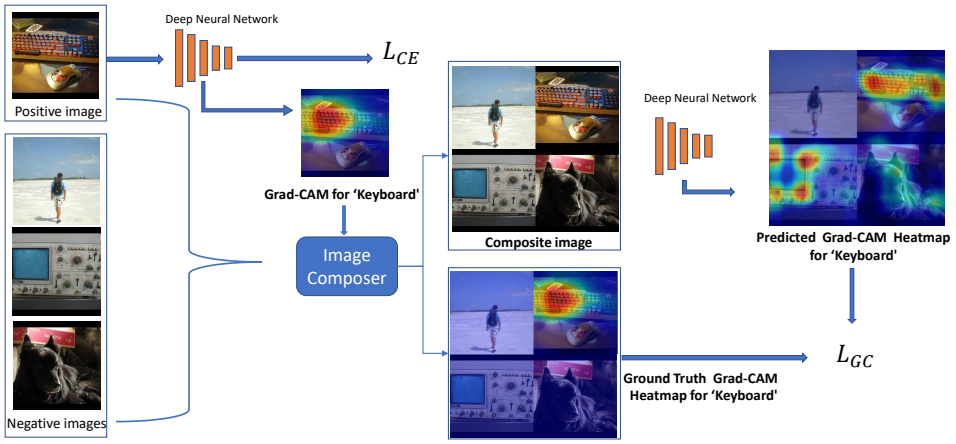

Figure 1: To encourage interpretation consistency, we randomly sample three distractor images for a given input image and feed all four to the Image Composer module which creates a 2×2 grid and places the positive image and the three negative images in random cells. We also feed the Grad-CAM interpretation mask for the ground truth category ('Keyboard') to the Image Composer to obtain the ground truth Grad-CAM mask of this composite image for the positive image category. The negative quadrants of this mask are set to zero. We then penalize the network if the Grad-CAM heatmap of the composite image for the positive image category does not match the ground truth Grad-CAM mask.

learning methods which have recently gained prominence.

Self-supervised learning: Recently, there is huge progress in developing self-supervised learning (SSL) algorithms which learn rich visual representations from unlabeled data. A class of SSL methods learn representations by enforcing some form of consistency that we know should exist in the visual feature space (Noroozi and Favaro 2016; Noroozi, Pirsiavash, and Favaro 2017; Zhang, Isola, and Efros 2016). Contrastive learning (Hadsell, Chopra, and LeCun 2006; He et al. 2019), that achieves state-of-the-art results, is an instance of this class. We develop a method inspired by these SSL methods to deal with the unlabeled ground truth interpretation masks.

In training the model, we apply a transformation in the input image for which we know the corresponding transformation in the interpretation space. Then, we design a loss to penalize deviation from the expected behaviour. More specifically, given an image, we compose a novel image using a 2×2 grid of images where the original image is on a random cell and the other three images are distractors that are randomly sampled from other categories. We know that the distractor images do not contain the category of interest (original image's category), so the interpretation of the model for the original category should be unaffected by the distractor images. Hence, we minimize the difference between interpretations of the original image and the composite image.

Moreover we know that there is not a clear evaluation protocol for network interpretation, so building better evaluation methods for interpretation is also an important research topic. We argue that the standard evaluation of localizing objects using interpretation is not a good fit as the interpretation does not need to focus on the entire object. Hence, we propose to evaluate the interpretation by measuring the percentage of heatmap (interpretation probability distribution) lying inside the object segmentation mask or bounding box. In other words, we prefer a model for which the interpretation does not use any pixel outside the human annotated object boundary in making the decision.

We design our method based on the Grad-CAM interpretation algorithm that is differentiable, and show that even though our model is tuned for consistency under Grad-CAM interpretation only, the trained model has consistent interpretations under a few other interpretation methods as well.

2 Related Work

2.1 Interpretability methods for black box models

Explainability of black box deep neural networks has been an active area of research in the past few years given the widespread adoption of deep neural networks for a variety of tasks. (Ribeiro, Singh, and Guestrin 2016) have shown that it is often very easy for machine learning models to pick up undesirable correlation artifacts during training which are often difficult to discover if we only rely upon prediction accuracy. Various approaches for detecting salient regions of an image has been proposed in the past few years (Zeiler and Fergus 2013; Simonyan, Vedaldi, and Zisserman 2013; Zhou et al. 2015b,a; Zhang et al. 2016). Zeiler and Fergus (Zeiler and Fergus 2013) introduced an approach that used the gradients of the class conditional output with respect to the input image and used spatial locations with large gradient magnitudes to obtain a saliency map corresponding to the class. (Springenberg et al. 2014; Sundararajan, Taly, and Yan 2017) build upon this work to obtain saliency maps which are sharper. (Ross, Hughes, and Doshi-Velez 2017) showed that input based gradient explanations match state of the art sample based explanations on several datasets. The authors also show that constraining the gradient explanations to be small in irrelevant areas using an annotation mask leads to improved gradient explanations, but comes at an added cost. Gradient-weighted Class Activation Mapping

(Grad-CAM) (Selvaraju et al. 2016) generalizes class activation maps (Zhou et al. 2015a) beyond global average pooling by using the gradients for a given class flowing into the final convolutional layer to obtain a coarse localization map highlighting the important regions in the image for predicting the class. Sanity checks (Adebayo et al. 2018) have shown that some of the existing methods produce saliency maps very similar to an edge detector and exhibit minimal sensitivity to randomization tests. Recently, (Subramanya, Pillai, and Pirsiavash 2019) showed that state-of-the-art interpretation algorithms might not always explain the true cause of a prediction and proposed evaluation metrics for assessing the reliability of network interpretation algorithms.

2.2 Explainable models

There have been recent works (Melis and Jaakkola 2018; Chen et al. 2019; Plumb et al. 2019) that attempt to train models that are explainable by definition. One such approach (Chen et al. 2019) introduces a new deep network architecture - ProtoPNet, that dissects a given input image by finding prototypical parts and makes a weighted combination of the prototypes to make a final classification. (Melis and Jaakkola 2018) introduce the notion of stability for explanations, i.e similar inputs should yield similar explanations. This is done by introducing a regularizer that encourages the model to behave like a linear model locally, but not globally and hence achieve interpretability without sacrificing accuracy. We build upon these ideas by incorporating a consistency constraint for the interpretation during the training phase of our model.

2.3 Self-supervised learning

Self-supervised learning deals with learning representations without explicit annotations by leveraging structural priors in the data. These priors are then used for automatically generating 'labels' for discriminative training. (Doersch, Gupta, and Efros 2015) introduced a *pretext* task that exploits structural properties in the visual domain to learn representations. Additional pretext tasks have since been proposed for images that exploit the spatial structure (Noroozi and Favaro 2016; Noroozi, Pirsiavash, and Favaro 2017; Noroozi et al. 2018), color information (Deshpande, Rock, and Forsyth 2015; Larsson, Maire, and Shakhnarovich 2016, 2017; Zhang, Isola, and Efros 2016), rotation (Gidaris, Singh, and Komodakis 2018), etc. We take inspiration from these techniques since we often lack annotations for network interpretation. In fact, interpretations can often be subjective and there is no clear definition of a ground truth for interpretation. (Wang et al. 2020) and (Guo et al. 2019) proposed training models to have consistent visual attention under different spatial transformations. Our method differs from these works by using distractor images while generating the composite training images and hence reducing the model's dependency on spurious contextual information.

3 Method

We first review Grad-CAM (Selvaraju et al. 2016) as a network interpretation method briefly since our model uses it in the learning.

Background on Grad-CAM visualization: Given an image x and a deep classification model f, we feed the image to the model and get a vector of logits y where y^t corresponds to the output for category t, and feeding y through a SoftMax operator produces the probability distribution over categories.

To interpret the model's decision, Grad-CAM of model f for category t produces a heatmap that highlights the regions of the image that lead the model to classify the image as category t. To do so, we pick a convolutional layer, e.g., conv5 in AlexNet, and calculate the derivative of the output y^t with respect to that layer and average it over space to get the importance of each channel of the convolutional layer:

$$\alpha_k^t = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^t}{\partial A_{ij}^k} \tag{1}$$

where A_{ij}^k is the activations of conv layer at channel k and location (i, j), Z is a normalizer, and α_k^t is the importance of channel k in making the decision. Then, we multiply each activation map by the corresponding importance, average them over the channels, and discard negative values:

$$gcam_{ij}^t = max(0, \sum_k \alpha_k^t A_{ij}^k) \tag{2}$$

Finally, to visualize it on image space, we up-sample it to the image size.

Element-wise Grad-CAM: In Eq. (1), the importance of each channel is averaged over the space to come up with one scalar for each channel. We believe this is sub-optimal as it does not consider the location information in the importance. Similar to (Saha et al. 2019), we fix this problem by removing the averaging and simply defining Grad-CAM as:

$$gcam_{ij}^t = max(0, \sum_k \frac{\partial y^t}{\partial A_{ij}^k} A_{ij}^k) \tag{3}$$

For input x and category y, we normalize Grad-CAM to sum to 1 and call it $g_x^y = \frac{gcam^y}{|gcam^y|_1}$.

Enforcing consistency in interpretation (GC): Fig. 1 shows our GC learning method. Given a training image x and its label y, we sample three random training images z_1, z_2, and z_3 that are labeled with any category except y. Then, we generate a twice larger image with a 2×2 grid and 4 cells. We place the above four images, x, z_1, z_2, and z_3 on random cells of the grid to come up with the composite image c:

$$c = comp(k, x, z_1, z_2, z_3)$$

where $comp(.)$ is the composition operator that simply concatenates the images in a 2×2 grid, and $k \in \{1...4\}$, that is picked randomly, is the cell index for the first image x. An example composite image is shown in the middle of Fig. 1.

We input x to the model f and calculate the Grad-CAM for the category y to get g_x^y. We also input the composite image c to the model f and calculate the Grad-CAM for the category y to get g_c^y.

Since we know that the distractor images z_i, do not contain the category y, we would like the network to not decide

on category y based on the regions of the distractor images. Hence, the heatmap for category y should be zero for the location of distractor images on the grid. Therefore, we generate the target heatmap for the composite image by placing g_x^y at the right cell and filling the other three cells with zeros.

$$\tilde{g}_c^y = comp(k, g_x^y, 0, 0, 0) \tag{4}$$

This can be seen as simply zero-padding the Grad-CAM visualization. Hence, we minimize the following loss function in training the model f:

$$\min_f \left(l_{ce}(f, x, y) + \lambda |g_c^y - \tilde{g}_c^y|_1 \right) \tag{5}$$

where the first term is simply the standard cross entropy loss that encourages the network to predict the correct label, the second term encourages consistency on the model interpretation, and λ is a hyper-parameter for trade-off between the two loss terms.

Note that Grad-CAM uses derivative of the model, so using its result in the loss function needs using double gradient of the model f in the optimization. This is straight forward in recent deep learning frameworks like PyTorch where the calculation of derivative itself is represented as a deep model.

ResNet with Global Max Pooling (GMP): Since ResNet architecture has a global average pooling over space in the last layer, it removes the location information and spreads out the explanation of a decision over the whole image. This is similar to the average pooling operation in standard Grad-CAM method that we removed in the element-wise Grad-CAM. Hence, to produce sharper interpretation and also enable the model to collect information from selective regions of the image rather than the whole image, we simply replace the global average pooling layer with a global max pooling layer. In the experiments, we show that this simple change produces more consistent interpretations without degrading the model's accuracy.

Interestingly, as a byproduct, we also show that this simple change in the model architecture makes the model more robust to FGSM adversarial attacks (Goodfellow, Shlens, and Szegedy 2014a). We believe this happens since the new model focuses on concise regions of the image rather than collecting information from all over the image. Then, the adversarial attack cannot fool the model to detect a fake "dog" by introducing weak features of dog over the whole image. Note that introducing a fake dog in a small region requires more perturbation that might be visible.

4 Experiments

4.1 Datasets

We perform all our experiments on ImageNet (Deng et al. 2009) and MS-COCO (Lin et al. 2014) datasets. For evaluation, we use the validation set of 50k images for ImageNet and $\approx$ 40k images for MS-COCO dataset.

4.2 Interpretation methods

We use the following network interpretation algorithms for our evaluations:

Grad-CAM: This is described in the method section.

Contrastive Top-down Attention (cMWP): (Zhang et al. 2016) introduced contrastive Marginal Winning Probability (cMWP) a stochastic Winner Takes All (WTA) formulation for CNN architectures for modelling the top-down attention for neural networks highlighting the most discriminative regions in an image used for classification. Here, the connections between activation neurons are considered to be *excitatory* if its weights are positive and *inhibitory* otherwise. During backpropagation, only the gradients for the excitatory connections are passed along the layers to obtain discriminative saliency maps for any layer.

FullGrad: FullGrad (Srinivas and Fleuret 2019) is another interpretation algorithm that creates an approximate saliency map visualization by aggregating the full-gradient components of the network for the given input. These full-gradient components are obtained by decomposing the output of a neural network into input sensitivity and per-neuron sensitivity components.

4.3 Evaluation metrics

We report model accuracy for all the models since we not only want the model to be interpretable, but also want it to perform well on the test set. Here, we describe the metrics we use for evaluating the visualization heatmaps:

Pointing Game (PG): This metric (Zhang et al. 2016) was introduced to quantitatively evaluate different interpretation algorithms using the ground truth semantic annotations. In this method, a given interpretation algorithm is used to compute a saliency map for each of the object classes in the input image. If the maximum point of this saliency map lies within the ground truth annotation mask of the object, we consider this as a hit, otherwise we consider this as a miss. Then the overall pointing game accuracy is computed as $\frac{\#Hits}{\#Hits+\#Misses}$. Similar to (Zhang et al. 2016), we dilate the object mask or bounding box by a margin before counting the number of hits to tolerate small misalignments. We use a margin of 15 pixels when images are of size 224×224.

Stochastic Pointing Game (SPG): One valid criticism for the pointing game metric is that it picks only the most salient point from the saliency map and hence only evaluates the *selectiveness* of a given interpretation algorithm for a model. This does not give us the full picture with respect to spurious correlations in the saliency map if its magnitude in the saliency map is not the maximum. Here, we are interested in evaluating not only the maximum point on the saliency map, but also the distribution of the saliency map across the image with respect to the annotation mask. To this end, we normalize the saliency map obtained from an interpretation algorithm to sum to 1 and treat it as a probability distribution over locations. We sample 100 spatial coordinates from this probability distribution, evaluate each sample similar to the pointing game, and report the average number of hits using the same 15 pixels margin as in PG. We call this method 'Stochastic Pointing Game' for which higher values are better. The value will be low if the saliency map highlights regions outside the object area.

Content Heatmap (CH): Next, we go a step further and

attempt to quantify the percentage of the heatmap that lies strictly within the object annotation mask for a given interpretation algorithm. If the model is expected to not rely on spurious contextual information for making classification decisions, we can assume that the percentage of the heatmap that lies inside the object annotation mask should be close to 1. Hence, we expect this metric to be high if the interpretation heatmap is mostly within the boundary of the object annotation mask.

Negative Quadrants Heatmap (NQH): Here, we use the same composite image creation method used to train our model with interpretation consistency and measure the percentage of the interpretation heatmap within the distractor quadrants. We expect this metric to be low since we know that the distractor quadrants do not contain the ground truth object by construction.

Note that CH is equal to the extreme case of SPG when we remove the tolerance margin and increase the number of trials to infinity. Also, PG is the other extreme of SPG when there is only one trial at the mode of the distribution. Hence, SPG is the intermediate point between two extreme cases.

4.4 Implementation details

We use AlexNet (Krizhevsky, Sutskever, and Hinton 2012), ResNet18 (He et al. 2015), and ResNet50 network architectures for all our experiments. We use PyTorch (Paszke et al. 2019) along with Nvidia Titan RTX and 2080Ti GPUs for training and evaluating our models. We maintain the resolution of images in the 2x2 composite image, so the composite image is 448x448 while the original image is 224x224. We use standard AlexNet and ResNet architectures in PyTorch which handle the change in input resolution with adaptive average pooling after the last convolutional layer. For training the models on the ImageNet dataset, we use SGD with a learning rate of 0.1 for ResNet18 and 0.01 for AlexNet decayed by 0.1 every 30 epochs. We set the λ hyperparameter in Eq (5) to 25 for the ImageNet experiments and 1 for the MS-COCO experiments respectively.

4.5 Results

We first describe the nomenclature of the methods: (1) GC refers to using Grad-CAM consistency loss in training. (2) GMP refers to using global max pooling in the ResNet architecture. We train the baseline models with ImageNet and MS-COCO datasets with standard cross entropy loss. For our Grad-CAM consistency method, we use our loss defined in Eq (5). For GC on MS-COCO dataset, we initialize from a model pretrained on ImageNet with our GC loss. The classification accuracy of the models are shown in Table 1 for ImageNet and in Table 2 for MS-COCO. Introduction of our GC loss and Max Pooling layer does not degrade the accuracy of the model significantly. Interestingly, our method improves the accuracy marginally on MS-COCO dataset. We did not aim for this improvement as we are mainly focused on improving the consistency of interpretation without losing much in accuracy.

Interpretation Consistency: Tables 1 and 2 show the results using the evaluation metrics from section 4.3 on the ImageNet and MS-COCO datasets respectively. For the stochastic pointing game evaluations, we report the mean and standard deviation values over 5 independent runs. Our method improves over the baseline model on all evaluation metrics for both datasets.

Consistency for Elementwise Grad-CAM: Since AlexNet doesn't have any pooling layer, we use Elementwise Grad-CAM without any change in the architecture. Table 5 shows that the AlexNet model trained with GC outperforms the baseline on both Grad-CAM and Elementwise Grad-CAM interpretation methods. More importantly, Elementwise Grad-CAM outperforms the standard Grad-CAM for both the baseline and GC. This empirically validate our hypothesis that standard Grad-CAM reduces the localization information. We use Elementwise Grad-CAM in all other experiments, but since the baseline ResNet models come with an average pooling layer, there is no difference between the original Grad-CAM and the Elementwise Grad-CAM.

Transfer across interpretation algorithms: To confirm that the improved interpretation for the model trained for interpretation consistency is not overfitted to the Grad-CAM, we also evaluate our model using other interpretation algorithms discussed in section 4.2. Tables 3 and 4 show that the improvements transfer to the other interpretation methods even though they are not used in the loss function. The learning always uses Grad-CAM in the loss.

Robustness against adversarial attacks: Unlike standard ResNet architecture, since the global max pooling layer will only focus on the most salient spatial location for each filter, we expect such a network to be more robust relative to the standard ResNet architecture with global average pooling. An adversarial attack on standard ResNet can add minor perturbations to multiple spatial locations and the average pooling operation will aggregate all these minor perturbations for the final classification layer; whereas, such an attack on the network using global max pooling will need a much larger magnitude ϵ adversarial attack to make the same adversarial attack successful. Table 6 compares the baseline ResNet50 with ResNet50 trained with GMP and also combined GMP+GC. Our results show that GMP+GC is able to withstand adversarial attacks of much larger magnitudes compared to the baseline. This is inline with our goal of reducing the effect of context in decision making, as using non-relevant information in making the decision opens the doors to the adversary to influence the decision by spreading the attack across the whole image area.

5 Qualitative Results

Fig. 2 shows some examples comparing the Grad-CAM visualization of our model with the baseline. We see that our method reduces the contextual correlation in the saliency map for a given object category in an image. We also compare our method with the ResNet18 model which uses Global Max Pooling instead of Global Average Pooling. We hypothesize that our method improves the resulting saliency map for a given object category in an image by focusing on the most *discriminative* part of the object.

Model		Top-1 Acc (%)	PG	SPG	CH (%)	NQH (%)
AlexNet	Baseline	56.51	72.80	53.45 ± 0.02	45.78	43.53
	GC	56.16	**73.70**	**61.15** ± 0.01	**48.10**	**33.94**
ResNet18	Baseline	69.43	79.80	60.50 ± 0.01	54.36	24.35
	GC	67.74	**80.00**	65.85 ± 0.01	57.73	**7.62**
	GMP	69.08	79.30	66.66 ± 0.01	62.89	38.96
	GMP + GC	69.02	79.60	**68.74 ± 0.01**	**65.35**	31.15
ResNet50	Baseline	76.13	80.0	60.95 ± 0.00	54.78	21.48
	GC	74.40	**80.30**	65.26 ± 0.00	59.42	**7.43**
	GMP	74.63	79.80	66.29 ± 0.00	54.23	33.91
	GMP + GC	74.14	79.60	**69.51 ± 0.00**	**59.70**	21.30

Table 1: Evaluation using Grad-CAM interpretation for AlexNet, ResNet18, and ResNet50 on the ImageNet validation set. Note that for the PG (Pointing Game), SPG (Stochastic Pointing Game) and CH (Content Heatmap), higher values are better; whereas, for NQH (Negative Quadrants Heatmap), lower values are better. We observe that when GC is combined with GMP, the resulting interpretation heatmaps show reduced influence of the regions outside the object annotation mask. We also observe a marginal drop in classification accuracy while yielding consistent interpretations.

Model		F1-PerClass	F1-Overall	PG	SPG	CH (%)	NQH (%)
ResNet18	Baseline	59.81	67.56	61.50	43.25 ± 0.01	29.15	29.27
	GC	61.50	68.59	63.80	44.26 ± 0.00	29.95	**28.35**
	GMP	61.19	68.41	63.10	45.62 ± 0.01	33.97	41.09
	GMP + GC	61.99	68.93	**66.10**	**46.88 ± 0.01**	**35.16**	40.17

Table 2: Evaluation using Grad-CAM interpretaion for ResNet18 on the MS-COCO validation set.

ResNet18	PG	SPG	CH (%)	NQH (%)
Baseline	63.30	45.68 ± 0.00	32.23	74.89
GC	65.60	48.54 ± 0.01	34.44	74.90
GMP	63.80	51.21 ± 0.01	37.54	75.21
GMP + GC	**65.60**	**52.22 ± 0.01**	**38.17**	**74.86**

Table 3: Evaluation using the Contrastive Excitation Backprop (Zhang et al. 2016) interpretation algorithm on the MS-COCO validation set

ResNet18	PG	SPG	CH (%)	NQH (%)
Baseline	59.90	18.99 ± 0.00	13.38	73.83
GC	64.90	19.46 ± 0.00	13.71	73.46
GMP	66.50	27.09 ± 0.01	**19.14**	68.64
GMP + GC	**68.70**	**27.18 ± 0.00**	**19.14**	**68.49**

Table 4: Evaluation using the FullGrad (Srinivas and Fleuret 2019) interpretation algorithm on MS-COCO validation set

AlexNet	PG		CH (%)	
	GCAM	Elementwise GCAM	GCAM	Elementwise GCAM
Baseline	72.8	74.0	45.78	46.79
GC	**73.7**	**76.6**	**48.10**	**55.20**

Table 5: Evaluation using the regular Grad-CAM and the Elementwise Grad-CAM interpretation on the ImageNet validation set for AlexNet.

ResNet50	ϵ for L_∞ adversarial attack using FGSM (Goodfellow, Shlens, and Szegedy 2014b)					
	1	2	8	16	32	64
Baseline	3.11	2.98	2.95	2.89	2.30	1.06
GMP	6.27	6.20	6.23	5.99	4.81	1.95
GMP + GC	**8.08**	**7.93**	**7.96**	**7.88**	**6.63**	**2.71**

Table 6: Evaluation of robustness for ResNet50 architecture. Our method when used with GMP on ResNet50 is able to withstand adversarial attacks of much larger magnitudes compared to the baseline.

6 Conclusion

We propose a novel learning method that enforces consistency in the interpretation of deep models. Since the ground-truth for the right interpretation is not known, we adopt ideas from self-supervised learning approaches that deal with unlabeled data for representation learning. We show that compared to standard models, our model focuses more on the objects of interest rather than the background regions. We experiment with two datasets and three network architectures and also show that our method transfers to other interpretation algorithms. Moreover, we introduce new evaluation metrics to evaluate the consistency of the interpretation.

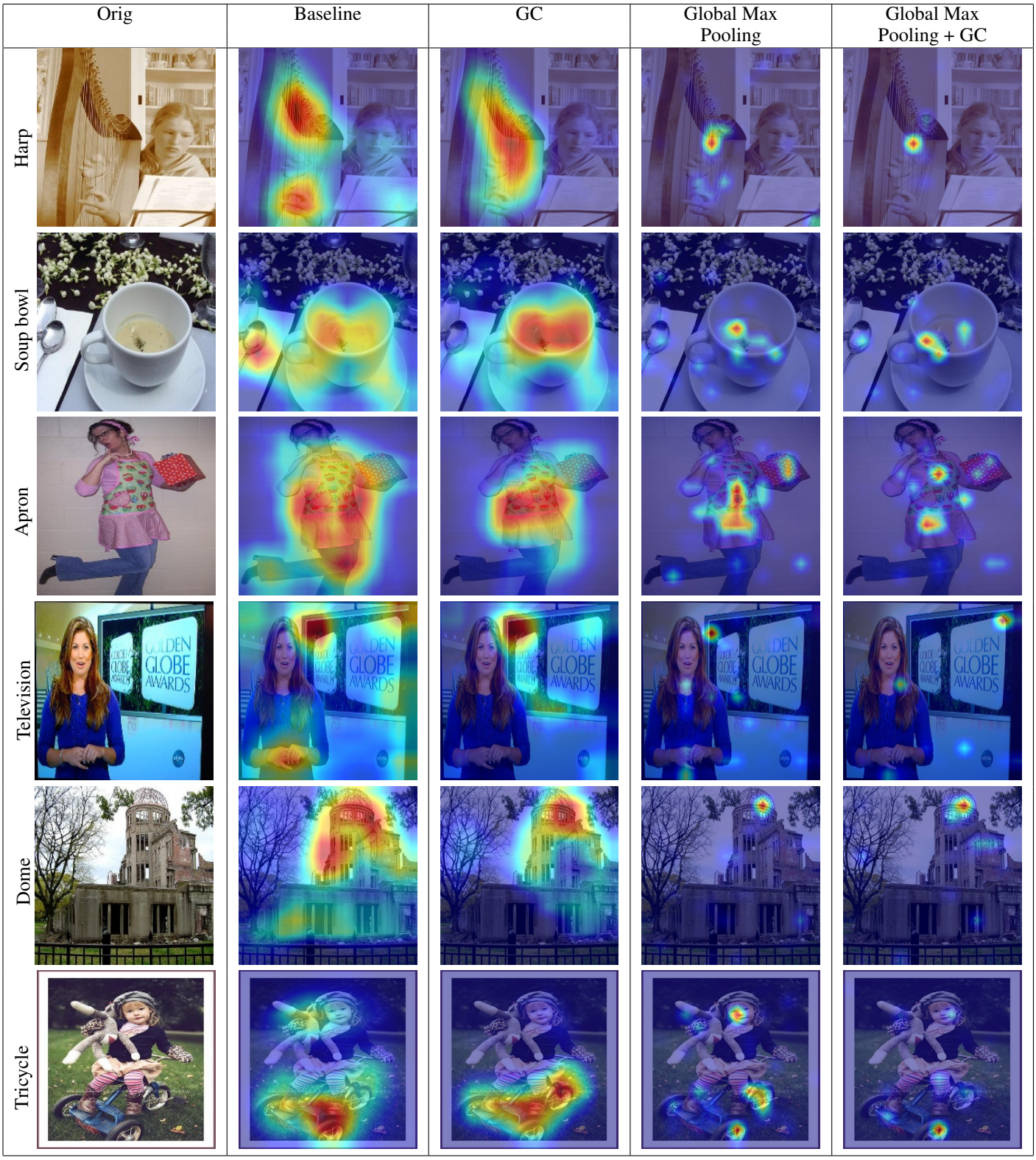

Figure 2: Grad-CAM visualization results for images from the ImageNet validation set using ResNet18. Our method has much better interpretation heatmaps compared to the baseline. In the first row, our method does not rely on the hand of the person to detect the 'harp' category, thereby reducing the importance of correlated contextual information. We also observe that the visualization maps for the Global Max Pooling model is much sharper and our method further improves upon this model.

Acknowledgment: This material is based upon work partially supported by the United States Air Force under Contract No. FA8750-19-C-0098, funding from NSF grant number 1845216, SAP SE, and Northrop Grumman. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the United States Air Force, DARPA, or other funding agencies.

7 Ethics Statement

Since our method works towards bringing the explanation of a network's decision making process closer to human priors, it can remove some of the ethical concerns by increasing transparency. However, similar to many other AI methods, our methods can be used by adversaries for unethical applications.

References

Adebayo, J.; Gilmer, J.; Muelly, M.; Goodfellow, I.; Hardt, M.; and Kim, B. 2018. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, 9505–9515.

Alexandrov, N. 2017. Explainable AI decisions for human-autonomy interactions. In *17th AIAA Aviation Technology, Integration, and Operations Conference*, 3991.

Caruana, R.; Lou, Y.; Gehrke, J.; Koch, P.; Sturm, M.; and Elhadad, N. 2015. Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission. In *KDD '15*.

Chen, C.; Li, O.; Tao, D.; Barnett, A.; Rudin, C.; and Su, J. K. 2019. This looks like that: deep learning for interpretable image recognition. In *Advances in Neural Information Processing Systems*, 8928–8939.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009. CVPR 2009.*, 248–255. IEEE.

Deshpande, A.; Rock, J.; and Forsyth, D. A. 2015. Learning Large-Scale Automatic Image Colorization. *2015 IEEE International Conference on Computer Vision (ICCV)* 567–575.

Doersch, C.; Gupta, A.; and Efros, A. A. 2015. Unsupervised Visual Representation Learning by Context Prediction. *2015 IEEE International Conference on Computer Vision (ICCV)* 1422–1430.

Gidaris, S.; Singh, P.; and Komodakis, N. 2018. Unsupervised Representation Learning by Predicting Image Rotations. In *International Conference on Learning Representations*. URL https://openreview.net/forum?id=S1v4N2l0-.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014a. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* .

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014b. Explaining and Harnessing Adversarial Examples. *CoRR* abs/1412.6572.

Guo, H.; Zheng, K.; Fan, X.; Yu, H.; and Wang, S. 2019. Visual attention consistency under image transforms for multi-label image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 729–739.

Hadsell, R.; Chopra, S.; and LeCun, Y. 2006. Dimensionality Reduction by Learning an Invariant Mapping. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* 2: 1735–1742.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. B. 2019. Momentum Contrast for Unsupervised Visual Representation Learning. *ArXiv* abs/1911.05722.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 770–778.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *NIPS*.

Larsson, G.; Maire, M.; and Shakhnarovich, G. 2016. Learning Representations for Automatic Colorization. In *ECCV*.

Larsson, G.; Maire, M.; and Shakhnarovich, G. 2017. Colorization as a Proxy Task for Visual Understanding. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 840–849.

Li, K.; Wu, Z.; Peng, K.-C.; Ernst, J.; and Fu, Y. 2019. Guided Attention Inference Network. *IEEE transactions on pattern analysis and machine intelligence* .

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer.

Melis, D. A.; and Jaakkola, T. 2018. Towards robust interpretability with self-explaining neural networks. In *Advances in Neural Information Processing Systems*, 7775–7784.

Noroozi, M.; and Favaro, P. 2016. Unsupervised Learning of Visual Represention by solving Jigsaw Puzzles. In *ECCV*.

Noroozi, M.; Pirsiavash, H.; and Favaro, P. 2017. Representation Learning by Learning to Count. *2017 IEEE International Conference on Computer Vision (ICCV)* 5899–5907.

Noroozi, M.; Vinjimoor, A.; Favaro, P.; and Pirsiavash, H. 2018. Boosting Self-Supervised Learning via Knowledge Transfer. In *CVPR 2018*.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Köpf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *NeurIPS*.

Plumb, G.; Al-Shedivat, M.; Xing, E. P.; and Talwalkar, A. 2019. Regularizing Black-box Models for Improved Interpretability. *ArXiv* abs/1902.06787.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *KDD '16*.

Ross, A. S.; Hughes, M. C.; and Doshi-Velez, F. 2017. Right for the Right Reasons: Training Differentiable Models by Constraining their Explanations. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, 2662–2670. doi:10.24963/ijcai.2017/371. URL https://doi.org/10.24963/ijcai.2017/371.

Saha, A.; Subramanya, A.; Patil, K. B.; and Pirsiavash, H. 2019. Adversarial Patches Exploiting Contextual Reasoning in Object Detection. *ArXiv* abs/1910.00068.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2016. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *2017 IEEE International Conference on Computer Vision (ICCV)* 618–626.

Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2013. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *CoRR* abs/1312.6034.

Singh, K. K.; Mahajan, D.; Grauman, K.; Lee, Y. J.; Feiszli, M.; and Ghadiyaram, D. 2020. Don't Judge an Object by Its Context: Learning to Overcome Contextual Bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11070–11078.

Springenberg, J. T.; Dosovitskiy, A.; Brox, T.; and Riedmiller, M. A. 2014. Striving for Simplicity: The All Convolutional Net. *CoRR* abs/1412.6806.

Srinivas, S.; and Fleuret, F. 2019. Full-Gradient Representation for Neural Network Visualization. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Subramanya, A.; Pillai, V.; and Pirsiavash, H. 2019. Fooling Network Interpretation in Image Classification. In *Proceedings of the IEEE International Conference on Computer Vision*, 2020–2029.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic Attribution for Deep Networks. In *ICML*.

Vellido, A. 2019. The importance of interpretability and visualization in machine learning for applications in medicine and health care. *Neural Computing and Applications* 1–15.

Wang, Y.; Zhang, J.; Kan, M.; Shan, S.; and Chen, X. 2020. Self-supervised Equivariant Attention Mechanism for Weakly Supervised Semantic Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12275–12284.

Zeiler, M. D.; and Fergus, R. 2013. Visualizing and Understanding Convolutional Networks. *ArXiv* abs/1311.2901.

Zhang, J.; Lin, Z. L.; Brandt, J.; Shen, X.; and Sclaroff, S. 2016. Top-Down Neural Attention by Excitation Backprop. In *ECCV*.

Zhang, R.; Isola, P.; and Efros, A. A. 2016. Colorful Image Colorization. In *ECCV*.

Zhou, B.; Khosla, A.; Lapedriza, À.; Oliva, A.; and Torralba, A. 2015a. Learning Deep Features for Discriminative Localization. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 2921–2929.

Zhou, B.; Khosla, A.; Lapedriza, À.; Oliva, A.; and Torralba, A. 2015b. Object Detectors Emerge in Deep Scene CNNs. In Bengio, Y.; and LeCun, Y., eds., *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. URL http://arxiv.org/abs/1412.6856.