

Visualizing Deep Networks by Optimizing with Integrated Gradients

Zhongang Qi,^{1,2} Saeed Khorram,¹ Li Fuxin¹

¹School of Electrical Engineering and Computer Science, Oregon State University

²Applied Research Center, PCG, Tencent

zhongangqi@tencent.com, {khorrams, lif}@oregonstate.edu

Abstract

Understanding and interpreting the decisions made by deep learning models is valuable in many domains. In computer vision, computing heatmaps from a deep network is a popular approach for visualizing and understanding deep networks. However, heatmaps that do not correlate with the network may mislead human, hence the performance of heatmaps in providing a faithful explanation to the underlying deep network is crucial. In this paper, we propose I-GOS, which optimizes for a heatmap so that the classification scores on the masked image would maximally decrease. The main novelty of the approach is to compute descent directions based on the integrated gradients instead of the normal gradient, which avoids local optima and speeds up convergence. Compared with previous approaches, our method can flexibly compute heatmaps at any resolution for different user needs. Extensive experiments on several benchmark datasets show that the heatmaps produced by our approach are more correlated with the decision of the underlying deep network, in comparison with other state-of-the-art approaches.

Introduction

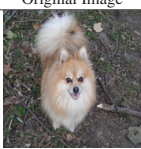
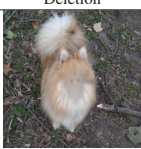
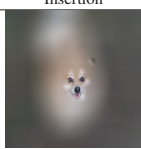



In recent years, there has been significant focus on explaining deep networks (Ribeiro, Singh, and Guestrin 2016; Lundberg and Lee 2017; Elenberg et al. 2017; Bau et al. 2017; Zhou et al. 2018; Zhang, Wu, and Zhu 2018; Alvarez-Melis and Jaakkola 2018). Explainability is important for humans to trust the deep learning model, especially in crucial decision-making scenarios. In the computer vision domain, one of the most important explanation techniques is the heatmap approach (Zeiler and Fergus 2014; Simonyan, Vedaldi, and Zisserman 2014; Selvaraju et al. 2017; Zhang et al. 2016), which focuses on generating heatmaps that highlight parts of the input image that are most important to the decision of the deep networks on a particular classification target. The heatmaps can be used to diagnose deep models to understand whether the models utilize the right contents to make the classification. This diagnosis is important when deep networks malfunction in high-stake cases, e.g. autonomous driving. In medical imaging

and other image domains that humans currently lack understanding, heatmaps can also be used to help humans gain further insights on which part of the images are important.

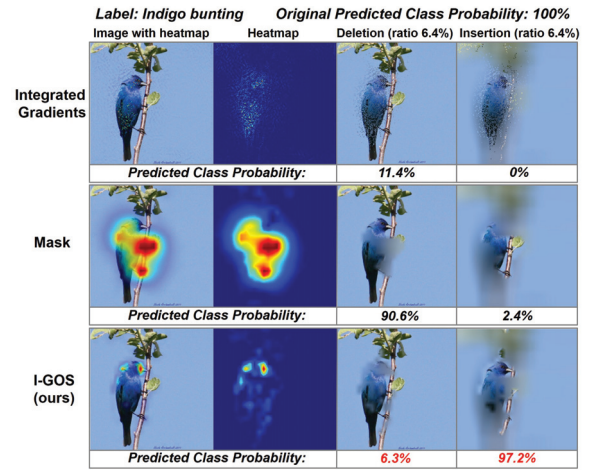
Some heatmap approaches achieve good visual qualities for human understanding, such as several one-step backpropagation-based visualizations including Guided Backpropagation (GBP) (Springenberg et al. 2015) and the deconvolutional network (DeconvNet) (Zeiler and Fergus 2014). These approaches utilize the gradient or variants of the gradient and backpropagate them back to the input image, in order to decide which pixels are more relevant to the change of the deep network prediction. However, whether they are actually correlated to the decision-making of the network is not that clear (Nie, Zhang, and Patel 2018). (Nie, Zhang, and Patel 2018) proves that GBP and DeconvNet are essentially doing (partial) image recovery, and thus generate more human-interpretable visualizations that highlight object boundaries, which do not necessarily represent what the model has truly learned.

An issue with these one-step approaches is that they only reflect infinitesimal changes of the prediction of a deep network. In the highly nonlinear function estimated by the deep network, such infinitesimal changes are not necessarily reflective of changes large enough to alter the decision of the model. (Petsiuk, Das, and Saenko 2018) proposed evaluation metrics based on masking the image with heatmaps and verifying whether the masking will indeed change deep network predictions. Ideally, if the highlighted regions for a category are removed from the image, the deep network should no longer predict that category. This is measured by the *deletion* metric. On the other hand, the network should predict a category only using the regions highlighted by the heatmap, which is measured by the *insertion* metric (Fig. 1).

If these are the goals of a heatmap, a natural idea would be to directly optimize them. The mask approach proposed in (Fong and Vedaldi 2017) generates heatmaps by solving an optimization problem, which aims to find the smallest and smoothest area that maximally decrease the output of a neural network, directly optimizing the *deletion* metric. It can generate very good heatmaps, but usually takes a long time to converge, and sometimes the optimization can be stuck in a bad local optimum due to the strong nonconvexity of the

Label	Original Image	Deletion	Insertion
259: 'Pomeranian'			
Predicted class probability	100%	4.8%	82.9%
Deletion or Insertion ratio	--	3.4%	2.3%
Label	Original Image	Deletion	Insertion
867: 'Trailer truck, tractor trailer, trucking rig, rig, articulated lorry, semi'			
Predicted class probability	99.7%	14.3%	86.2%
Deletion or Insertion ratio	--	8.0%	3.1%

(a)



(b)

Figure 1: (a) Examples generated by I-GOS in the deletion and insertion tasks using VGG19 as the baseline model. Heatmaps can be verified by testing the CNN on (multiple) deletion images (column 3), which blur the most highlighted areas on the heatmap, and (multiple) insertion images (column 4), which blur areas not highlighted on the heatmap. (b) The first two rows show that Integrated Gradients, Mask may fail on these evaluations. In the third row, I-GOS performs significantly better since the CNN no longer classifies the image to the same category with a small deleted area (column 3), and classifies the image correctly with only few pixels revealed (column 4), showing the correlation between the I-GOS heatmap and CNN decision making. For all approaches the same amount of pixels (6.4% in this figure) were blurred/revealed. (Best viewed in color)

solution space (Fig. 1).

In this paper, we propose a novel visualization approach I-GOS (Integrated-Gradients Optimized Saliency) which utilizes an idea called integrated gradients to improve the mask optimization approach in (Fong and Vedaldi 2017). The integrated gradients approach explicitly find a *baseline* image with very low prediction confidence – a completely grey or highly blurred image – then compute a straight line between the original image and this baseline. The gradients on images on this line are then integrated (Sundararajan, Taly, and Yan 2017). The idea is that the direction provided by the integrated gradients may lead better towards the global optimum than the normal gradient which may tend to lead to local optima. However, the original integrated gradient (Sundararajan, Taly, and Yan 2017) paper is a one-step visualization approach and generate diffuse heatmaps difficult for human to understand (Fig. 1). In this paper, we replace the gradient in mask optimization with the integrated gradients. Due to the high cost of computing the integrated gradients, we employ a line-search based gradient-projection method to maximally utilize each computation of the integrated gradients. We also utilize some empirical perturbation strategies to avoid the creation of adversarial masks. In the end, our approach generates better heatmaps (Fig. 1) and utilizes less computational time than the original mask optimization, as line search is more efficient in finding appropriate step sizes, allowing significantly less iterations to be used. We highlight our contributions as follows:

- (1) We developed a novel heatmap visualization approach I-GOS, which optimizes a mask using the integrated gradients as descent steps.

- (2) Through regularization and perturbation we better avoided generating adversarial masks at higher resolutions, enabling more detailed heatmaps that are more correlated with the decision-making of the model.
- (3) Extensive evaluations show that the proposed approach performs better than the state-of-the-art approaches, especially in the *insertion* and *deletion* metrics.

Related Work

There are several different types of the visualization techniques for generating heatmaps for a deep network. We classify them into one-step backpropagation-based approaches (Zeiler and Fergus 2014; Simonyan, Vedaldi, and Zisserman 2014; Springenberg et al. 2015; Shrikumar et al. 2016; Sundararajan, Taly, and Yan 2017; Bach et al. 2015; Zhang et al. 2016; Selvaraju et al. 2017), and perturbation-based approaches, e.g., (Zhou et al. 2014; Dabkowski and Gal 2017; Fong and Vedaldi 2017; Petsiuk, Das, and Saenko 2018).

The basic idea of one-step backpropagation-based visualizations is to backpropagate the output of a deep neural network back to the input space using the gradient or its variants. DeconvNet (Zeiler and Fergus 2014), Saliency Maps (Simonyan, Vedaldi, and Zisserman 2014), and GBP (Springenberg et al. 2015) are similar approaches, with the difference among them in the way they deal with the ReLU layer. LRP (Bach et al. 2015) and DeepLIFT (Shrikumar et al. 2016) compute the contributions of each input feature to the prediction. Excitation BP (Zhang et al. 2016) passes along top-down signals downwards in the network hierarchy via a probabilistic Winner-Take-All process. GradCAM (Selvaraju et al. 2017) uses the gradients of a target concept,

flowing only into the final convolutional layer to produce a coarse localization map. (Ancona et al. 2017) analyzes various backpropagation-based methods, and provides a unified view to explore the connections among them.

Perturbation-based methods first perturb parts of the input, and then run a forward pass to see which ones are most important to preserve the final decision. The earliest approach (Zhou et al. 2014) utilized a grey patch to occlude part of the image. This approach is direct but very slow, usually taking hours for a single image (Ancona et al. 2017). An improvement is to introduce a mask, and solve for the optimal mask as an optimization problem (Dabkowski and Gal 2017; Fong and Vedaldi 2017). (Dabkowski and Gal 2017) develop a trainable masking model that can produce the masks in a single forward pass. However, it is difficult to train a mask model, and different models need to be trained for different networks. (Fong and Vedaldi 2017) directly solves the optimization, and find the mask iteratively. Instead of only occluding one patch of the image, RISE (Petsiuk, Das, and Saenko 2018) generates thousands of randomized input masks simultaneously, and averages them by their output scores. However, it consumes significant time and GPU memory.

Another seemingly related but different domain is the saliency map from human fixation (Johnson and Subha 2017). Fixation Prediction (Kruthiventi et al. 2016; Kummerer et al. 2017) aims to identify the fixation points that human viewers would focus on at first glance of a given image, usually by training a network to predict those fixation points. This is different from deep explanation because deep models may use completely different mechanisms to classify from humans, hence human fixations should not be used to train or evaluate heatmap models.

Model Formulation

Gradient and Mask Optimization

Gradient and its variants are often utilized in visualization tools to demonstrate the importance of each dimension of the input. Its motivation comes from the linearization of the model. Suppose a black-box deep network f predicts a score $f_c(I)$ on class c (usually the logits of a class before the softmax layer) from an image I . Assume f is smooth at the current image I_0 , then a local approximation can be obtained using the first-order Taylor expansion:

$$f_c(I) \approx f_c(I_0) + \langle \nabla f_c(I_0), I - I_0 \rangle, \quad (1)$$

The gradient $\nabla f_c(I_0)$ is indicative of the local change that can be made to $f_c(I_0)$ if a small perturbation is added to it, and hence can be visualized as an indication of salient image regions to provide a local explanation for image I_0 (Simonyan, Vedaldi, and Zisserman 2014). In (Shrikumar et al. 2016), the heatmap is computed by multiplying the gradient feature-wise with the input itself, i.e., $\nabla f_c(I_0) \odot I_0$, to improve the sharpness of heatmaps.

However, gradient only illustrates the infinitesimal change of the function $f_c(I)$ at I_0 , which is not necessarily indicative of the salient regions that lead to a significant change on $f_c(I)$, especially when the function is highly nonlinear. What we would expect is that the heatmaps indicate the areas that would really change the classification

result significantly. In (Fong and Vedaldi 2017), a perturbation based approach is proposed which introduces a mask M as the heatmap to perturb the input I_0 . M is optimized by solving the following objective function:

$$\begin{aligned} \operatorname{argmin}_M F_c(I_0, M) &= f_c(\Phi(I_0, M)) + g(M), \\ \text{where } g(M) &= \lambda_1 \| \mathbf{1} - M \|_1 + \lambda_2 \text{TV}(M), \\ \Phi(I_0, M) &= I_0 \odot M + \tilde{I}_0 \odot (\mathbf{1} - M), \quad \mathbf{0} \leq M \leq \mathbf{1}, \end{aligned} \quad (2)$$

In (2), M is a matrix which has the same shape as the input image I_0 and whose elements are all in $[0, 1]$; \tilde{I}_0 is a baseline image with the same shape as I_0 , which should have a low score on the class c , $f_c(\tilde{I}_0) \approx \min_I f_c(I)$, and in practice either a constant image, random noise, or a highly blurred version of I_0 . This optimization seeks to find a deletion mask that significantly decreases the output score $f_c(\Phi(I_0, M))$, i.e., $f_c(I_0 \odot M + \tilde{I}_0 \odot (\mathbf{1} - M)) \ll f_c(I_0)$ under the regularization of $g(M)$. $g(M)$ contains two regularization terms, with the first term on the magnitude of M , and the second term a total-variation (TV) norm (Fong and Vedaldi 2017) to make M more piecewise-smooth.

Although this approach of optimizing a mask performs significantly better than the gradient method, there exist inevitable drawbacks when using a traditional first-order algorithm to solve the optimization. First, it is slow, usually taking hundreds of iterations to obtain the heatmap for each image. Second, since the model f_c is highly nonlinear in most cases, optimizing (2) may only achieve a local optimum, with no guarantee that it indicates the right direction for a significant change related to the output class. Fig. 1 and Fig. 3 show some heatmaps generated by the mask approach.

Integrated Gradients

Note that the problem of finding the mask is not a conventional non-convex optimization problem. For $F_c(I_0, M) = f_c(I_0, M) + g(M)$, we (approximately) know the global minimum (or, at least a reasonably small value) of $f_c(I_0, M)$ in a baseline image \tilde{I}_0 , which corresponds to $M = \mathbf{0}$. The integrated gradients (Sundararajan, Taly, and Yan 2017) consider the straight-line path from the baseline \tilde{I}_0 to the input I_0 . Instead of evaluating the gradient at the provided input I_0 only, the integrated gradients would be obtained by accumulating all the gradients along the path:

$$IG_i(I_0) = (I_0^i - \tilde{I}_0^i) \cdot \int_{\alpha=0}^1 \frac{\partial f_c(\tilde{I}_0 + \alpha(I_0 - \tilde{I}_0))}{\partial I_0^i} d\alpha, \quad (3)$$

where $IG(I_0) = \nabla_{I_0}^{IG} f_c(I_0)$ is the integrated gradients of f_c at I_0 ; i represents the i -th pixel.

In practice, the integral in (3) is approximated via a summation. We sum the gradients at points occurring at sufficiently small intervals along the straight-line path from the input M to a baseline $\tilde{M} = \mathbf{0}$:

$$\nabla^{IG} f_c(M) = \frac{1}{S} \sum_{s=1}^S \frac{\partial f_c(\Phi(I_0, \frac{s}{S} M))}{\partial M}, \quad (4)$$

where S is a constant, usually 20. However, (Sundararajan, Taly, and Yan 2017) only proposed to use integrated gradients as a one-step visualization method, and the heatmaps

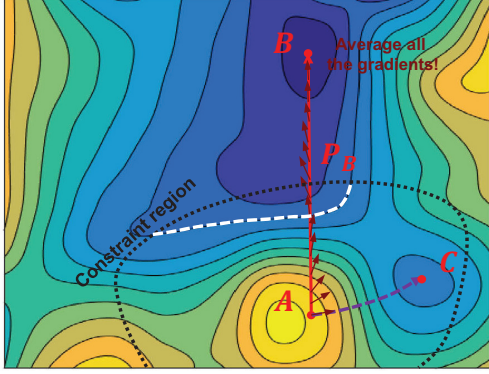


Figure 2: (Best viewed in color) Suppose we are optimizing in a region with a starting point A , a local optimum C , and a baseline B which is the unconstrained global optimum; the area within the black dashed line is the constraint region which is decided by the constraint terms $g(I, M)$ and the bound constraints $\mathbf{0} \leq M \leq \mathbf{1}$, we may find a better solution by always moving towards B rather than following the gradient and end up at C .

generated by the integrated gradients are still diffuse. Fig. 1 and Fig. 3 show some heatmaps generated by the integrated gradients approach where a grey zero image is utilized as the baseline. We can see that the integrated gradient contains many false positives in the area wherever the pixels have a large value of $I_0^i - \tilde{I}_0^i$ (either the white or the black pixels).

Integrated Gradients Optimized Heatmaps

We believe the above two approaches can be combined for a better heatmap approach. The integrated gradient naturally provides a better direction than the gradient in that it points more directly to the global optimum of a part of the objective function. One can view the convex constraint function $g(M)$ as equivalent to the Lagrangian of a constrained optimization approach with constraints $\|\mathbf{1} - M\|_1 \leq B_1$ and $TV(M) \leq B_2$, B_1 and B_2 being positive constants, hence consider the optimization problem (2) to be a constrained minimization problem on $f_c(\Phi(I_0, M))$. In this case, we know the unconstrained solution in $M = \mathbf{0}$ is outside the constraint region. We speculate that an optimization algorithm may be better than gradient descent if it directly attempts to move to the unconstrained global optimum.

To illustrate this, Fig. 2 shows a 2D optimization with a starting point A , a local optimum C , and a baseline B . The area within the black dashed line is the constraint region which is decided by the constraint function $g(M)$ and the boundary of M . A first-order algorithm will follow the gradient descent direction (the purple line) to the local optimum C ; while the integrated gradients computed along the path P_B from A to the baseline B may enable the optimization to reach an area better than C within the constraint region. We can see that the integrated gradients with an appropriate baseline have a global view of the space and may generate a better descent direction. In practice, the baseline does not need to be the global optimum. A good baseline near the

global optimum could still improve over the local optimum achieved by gradient descent.

Hence, we utilize the integrated gradients to substitute the gradient of the partial objective $f_c(M)$ in optimization (2), and introduce a new visualization method called Integrated-Gradient Optimized Saliency (I-GOS). For the regularization terms $g(M)$ in optimization (2), we still compute the partial (sub)gradient with respect to M :

$$\nabla g(M) = \lambda_1 \cdot \frac{\partial \|\mathbf{1} - M\|_1}{\partial M} + \lambda_2 \cdot \frac{\partial TV(M)}{\partial M}, \quad (5)$$

The total (sub)gradient of the optimization for M at each step is the combination of the integrated gradients for the $f_c(M)$ and the gradients of the regularization terms $g(M)$:

$$TG(M) = \nabla^{IG} f_c(M) + \nabla g(M), \quad (6)$$

Note that this is no longer a conventional optimization problem, since it contains 2 different types of gradients. The integrated gradients are utilized to indicate a direction for the partial objective $f_c(M)$; the gradients of the $g(M)$ are used to regularize this direction and prevent it to be diffuse.

Computing the step size

Since the time complexity of computing $\nabla^{IG} f_c(M)$ is high, we utilize a backtracking line search method and revise the Armijo condition (Nocedal and Wright 2000) to help us compute the appropriate step size for the total gradient $TG(M)$ in formula (6). The Armijo condition tries to find a step size such that:

$$f(M_k + \alpha_k \cdot d_k) - f(M_k) \leq \alpha_k \cdot \beta \cdot \nabla f(M_k)^T d_k, \quad (7)$$

where d_k is the descent direction; α_k is the step size; β is a parameter in $(0, 1)$; $\nabla f(M_k)$ is the gradient of f at point M_k .

The descent direction d_k for our algorithm is set to be the inverse direction of the total gradient $TG(M_k)$. However, since $TG(M_k)$ contains the integrated gradients, it is uncertain whether $\nabla F_c(M_k)^T d_k = -\nabla F_c(M_k)^T TG(M_k)$ is negative or not. Hence, we replace $\nabla F_c(M_k)$ with $TG(M_k)$ and obtain a revised Armijo condition as follows:

$$F_c\left(M_k - \alpha_k \cdot TG(M_k)\right) - F_c(M_k) \leq -\alpha_k \cdot \beta \cdot TG(M_k)^T TG(M_k), \quad (8)$$

The detailed backtracking line search works as follows:

- (1) Initialization: set the values of the parameter β , a decay η , an upper bound α_u and a lower bound α_l for the step size; let $j = 0$, and $\alpha^0 = \alpha_u$;
- (2) Iteration: if α^j satisfies condition (8), or $\alpha_j \leq \alpha_l$, end iteration; else, let $\alpha^{j+1} = \alpha^j \eta$, $j = j + 1$, test condition (8) again with $P_{[0,1]}(M_k - \alpha_k \cdot TG(M_k))$, where $P_{[0,1]}(M)$ clips the mask values to the closed interval $[0, 1]$;
- (3) Output: if $\alpha^j \leq \alpha_l$, the step size α_k for $TG(M_k)$ equals to the lower bound α_l ; else, $\alpha_k = \alpha^j$

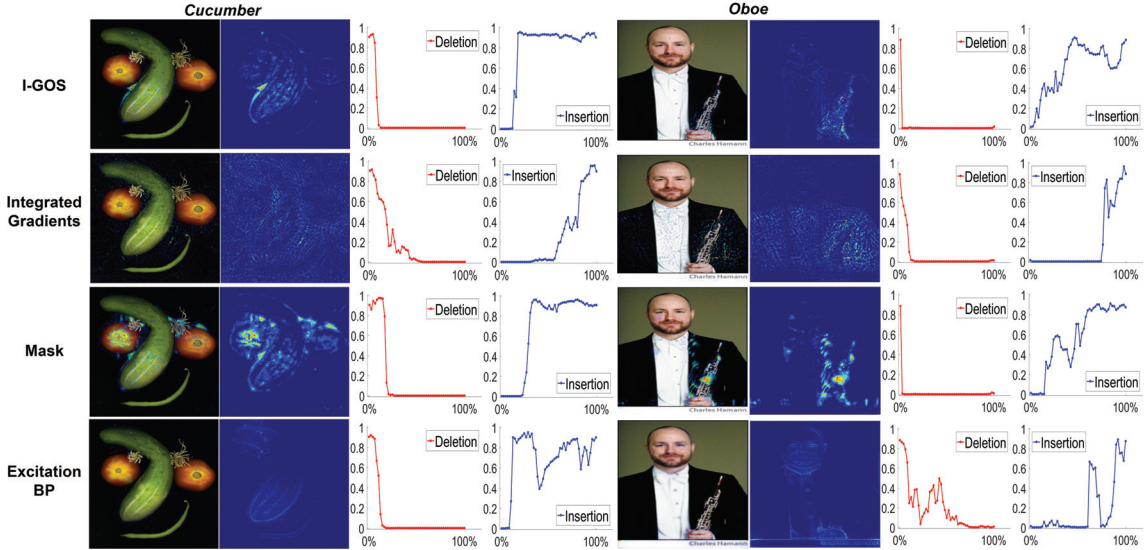


Figure 3: Different heatmap approaches at 224×224 resolution. The red plot illustrates how the CNN predicted probability drops with more areas masked, and the blue plot illustrates how the prediction increases with more areas revealed. The x axis for the red/blue plot represents the percentage of pixels masked/revealed; and the y axis represents the predicted class probability. One can see with I-GOS the red curve drops earlier and the blue plot increase earlier, leading to more area under the insertion curve (insertion metric) and less area under the deletion curve (deletion metric). (Best viewed in color)

A projection step is needed in the iteration because the mask M_k is bounded by the closed interval $[0, 1]$. Since we have an integrated gradient in $TG(M)$, a large upper bound α_u and a small β are needed in order to obtain a large step that satisfies condition (8), similar to satisfying the Goldstein conditions for convergence in conventional Armijo-Goldstein line search.

Note that we cannot prove the convergence properties of the algorithm in non-convex optimization. However, the integrated gradient reduces to a scaling on the conventional gradient in a quadratic function (see supplementary material). In practice, it converges much faster than the original mask approach in (Fong and Vedaldi 2017) and we have never observed it diverging, although in some cases we do note that even with this approach the optimization stops at a local optimum. With the line search, we usually only run the iteration for 10 – 20 steps. Intuitively, the irrelevant parts of the integrated gradients are controlled gradually by the regularization function $g(M)$ and only the parts that truly correlate with output scores would remain in the final heatmap.

Avoiding adversarial examples

Since the mask optimization (2) is similar to the adversarial optimization (Szegedy et al. 2014; Goodfellow et al. 2014) except the TV term, it is concerning whether the solution would merely be an adversarial attack to the original image rather than explaining the relevant information. An adversarial example is essentially a mask that drives the image off the natural image manifold, hence the approach in (Fong and Vedaldi 2017) utilize a blurred version of the original image as the baseline to avoid creating strong adversarial gradients off the image manifold. We follow (Fong and Vedaldi 2017)

and also use a blurred image as the baseline. The total variation constraints also defeats adversarial masks by making the mask piecewise-smooth. We also added other methods to avoid finding an adversarial perturbation:

Algorithm 1: I-GOS

Optimization objective: formula (9);
Initialization: set $M_0 = 1$;
while not converged and within the maximum steps **do**
 Add different random noise n_s to I_0 when computing the integrated gradient: $\nabla^{IG} f_c(M_k) = \frac{1}{S} \sum_{s=1}^S \partial f_c(\Phi(I_0 + n_s, \frac{s}{S} \text{up}(M_k))) / \partial M_k$;
 Compute the total (sub)gradient $TG(M_k)$ of the optimization for M_k using formula (6);
 Compute the step size α_k using the backtracking line search algorithm;
 Update: $M_{k+1} = M_k - \alpha_k \cdot TG(M_k)$;
end

(1) When computing the integrated gradients using formula (4), we add different random noise n_s to I_0 at each point along the straight-line path.

(2) When computing a mask M whose resolution is smaller than that of the input image I_0 , we upsample it before perturbing the input I_0 , and rewrite formula (2) as:

$$M^* = \operatorname{argmin} f_c(\Phi(I_0, \text{up}(M))) + \lambda_1 \|1 - M\|_1 + \lambda_2 \text{TV}(M), \quad (9)$$

where $\text{up}(M)$ upsamples M to the original resolution with bilinear upsampling. The lower the resolution of M , the smoother generated heatmap is.

Table 1: Evaluation in terms of deletion (lower is better) and insertion (higher is better) scores on the ImageNet dataset using the VGG19 model. GradCam can only generate 14×14 heatmaps for VGG19; RISE and Integrated Gradients can only generate 224×224 heatmaps

	224×224		112×112		28×28		14×14	
	Deletion	Insertion	Deletion	Insertion	Deletion	Insertion	Deletion	Insertion
Excitation BP (Zhang et al. 2016)	0.2037	0.4728	0.2053	0.4966	0.2202	0.5256	0.2328	0.5452
Mask (Fong and Vedaldi 2017)	0.0482	0.4158	0.0728	0.4377	0.1056	0.5335	0.1753	0.5647
GradCam (Selvaraju et al. 2017)	--	--	--	--	--	--	0.1527	0.5938
RISE (Petsiuk, Das, and Saenko 2018)	0.1082	0.5139	--	--	--	--	--	--
Integrated Gradients (Sundararajan, Taly, and Yan 2017)	0.0663	0.2551	--	--	--	--	--	--
I-GOS (ours)	0.0336	0.5246	0.0609	0.5153	0.0899	0.5701	0.1213	0.6387

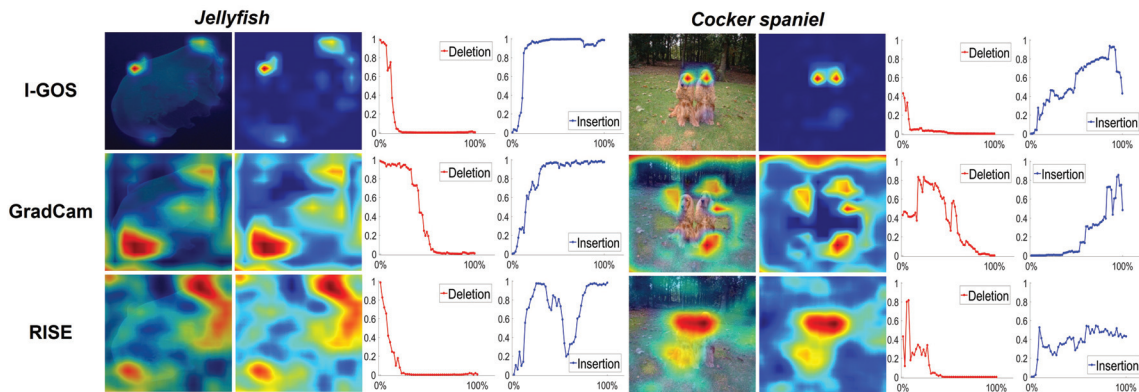


Figure 4: Comparisons between GradCam, RISE, and I-GOS, see Fig. 3 caption for explanations of the meaning of the curves.

Whether a mask is adversarial can be evaluated using the *insertion metric*, detailed in the experiments section. We summarize an overview of the proposed I-GOS approach in Algorithm 1.

Experiments

Evaluation Metrics and Parameter Settings

Although many recent work focus on explainable machine learning, there is still no consensus about how to measure the explainability of a machine learning model. For the heatmaps, one of the important issues is whether we are explaining the image with human’s understanding or the deep model’s perspective. A common pitfall is to try to use human’s understanding to explain the deep model, e.g. the pointing game (Zhang et al. 2016), which measures the ability of a heatmap to focus on the ground truth object bounding box. However, there are plenty of evidences that deep learning sometimes uses background context for object classification which would invalidate pointing game evaluations. Many heatmap papers show appealing images which look plausible to humans, but (Nie, Zhang, and Patel 2018) points out they could well be just doing partial image recovery and boundary detection, hence generate human-interpretable results that do not correlate with network prediction. Hence, it is important to utilize objective metrics that have causal effects on the network prediction for the evaluation.

We follow (Petsiuk, Das, and Saenko 2018) to adopt *deletion* and *insertion* as better metrics to evaluate different heatmap approaches. In the *deletion* metric, we remove N

pixels (dependent on the resolution of the mask) most highlighted by the heatmap each time from the original image iteratively until no pixel is left, and replace the removed ones with the corresponding pixels from the baseline image. The deletion score is the area under the curve (AUC) of the classification scores after softmax (Petsiuk, Das, and Saenko 2018) (red curve in Fig. 3-5). For the *insertion* metric, we replace N most highlighted pixels from the baseline image with the ones from the original image iteratively until no pixel left (blue curve in Fig.3-6). The insertion score is also the AUC of the classification scores for all the replaced images. In the experiments, we generate heatmaps with different resolutions, including 224×224 , 112×112 , 28×28 , 14×14 , and 7×7 . And we compute the deletion and insertion scores by replacing pixels based on generated heatmaps at the original resolutions before upsampling.

The intuition behind the *deletion* metric is that the removal of the pixels most relevant to a class will cause the prediction confidence to drop sharply. This is similar to the optimization goal in eq. (2). Hence, only utilizing the *deletion* metric is not satisfactory enough since adversarial attacks can also achieve a quite good performance. The intuition behind the *insertion* metric is that only keeping the most relevant pixels will retain the original score as much as possible. Since adversarial masks usually only optimize the deletion metric, it often use irrelevant parts of the image to drop the prediction score. Thus, if only those parts are revealed to the model, usually the model would not make a confident prediction on the original class, hence a low insertion score. Therefore, a good *insertion* metric indicates

Table 2: Comparative evaluation in terms of deletion (lower is better) and insertion (higher is better) scores on ImageNet using ResNet50 as the base model. GradCam can only generate 7×7 heatmaps for ResNet50; RISE and Integrated Gradients only generate 224×224 heatmaps

	224×224		112×112		28×28		14×14		7×7	
	Deletion	Insertion	Deletion	Insertion	Deletion	Insertion	Deletion	Insertion	Deletion	Insertion
Mask (Fong and Vedaldi 2017)	0.0468	0.4962	0.0746	0.5090	0.1151	0.5559	0.1557	0.5959	0.2259	0.6003
GradCam (Selvaraju et al. 2017)	---	---	---	---	---	---	---	---	0.1675	0.6521
RISE (Petsiuk, Das, and Saenko 2018)	0.1196	0.5637	---	---	---	---	---	---	---	---
Integrated Gradients (Sundararajan, Taly, and Yan 2017)	0.0907	0.2921	---	---	---	---	---	---	---	---
I-GOS (ours)	0.0420	0.5846	0.0704	0.5943	0.1059	0.5986	0.1387	0.6387	0.1607	0.6632

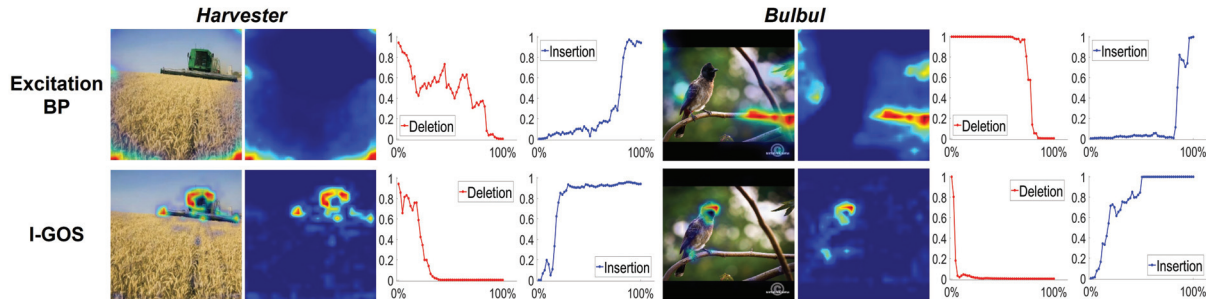


Figure 5: Comparison between Excitation BP and I-GOS at resolution 28×28 . See Fig. 3 for explanations of the figures

a non-adversarial mask. However, only using the insertion metric would not identify blurry masks (e.g. Fig. 4), hence the *deletion-insertion* metrics should be considered jointly.

For the deletion and insertion task, we utilize the pretrained VGG19 (Simonyan and Zisserman 2015) and Resnet50 (He et al. 2016) networks from the PyTorch model zoo to test 5,000 randomly selected images from the validation set of ImageNet (Russakovsky et al. 2015). In Eq. (8), $\beta = 0.0001$. λ_1 and λ_2 in Eq. (9) were fixed across all experiments under the same heatmap resolution.

We downloaded and ran the code for most baselines, except for (Sundararajan, Taly, and Yan 2017) which we implemented. All baselines were tuned to best performances. For RISE, we followed (Petsiuk, Das, and Saenko 2018) to generate 4,000 7×7 random samples for VGG, and 8,000 7×7 random samples for ResNet. For all experiments we used the same pre-/post-processing with the same baseline image \tilde{I}_0 . (Petsiuk, Das, and Saenko 2018) used a less blurred image for insertion and a grey image for deletion. Since we found the blurriness in (Petsiuk, Das, and Saenko 2018) was not always enough to get the CNN to output 0 confidence, we used a more blurred image for both insertion and deletion, hence the insertion and deletion scores for RISE are bit different in our paper compared with theirs.

Results and Discussions

Deletion and Insertion: Table 1 and 2 show the comparative evaluations of I-GOS with other state-of-the-art approaches in terms of the *deletion* and *insertion* metrics on the ImageNet dataset using VGG19 and ResNet50 as the baseline model, respectively. From Table 1 and 2 we observe that our proposed approach I-GOS performs better than all baselines in both deletion and insertion scores for heatmaps with

all different resolutions.

Integrated Gradients obtains the worst insertion score among all the approaches, which indicates that it indeed contains lots of pixels uncorrelated with the classification, as in the *Cucumber* and *Oboe* examples in Fig. 3. Excitation BP sometimes fires on irrelevant parts of the image as argued in (Nie, Zhang, and Patel 2018). Thus, it performs the worst in the deletion task. GradCAM and RISE also suffer on the deletion score maybe because of the randomness on the masks they generate, which sometimes fixate on random background regions irrelevant to classification. Fig. 3-5 shows some visual comparisons between our approach and baselines at various resolutions. The reason of insertion curve going down and up is that sometimes part of the image that contains features that are indicative of other classes could be inserted, which could increase the activation for other classes, potentially driving down the softmax probability for the current class.

Note that one advantage of our approach compared to the previous best RISE and GradCAM is the flexibility in terms of resolutions. RISE and Integrated Gradients can only generate 224×224 heatmaps. GradCam can only generate 14×14 heatmap on VGG19, and 7×7 heatmap on Resnet50, respectively. Our approach is better than them at their resolutions, but also offers the flexibility to use other resolutions. High resolutions are necessary especially when the image has thin parts (e.g. Fig. 6), however may be less visually appealing since the masked pixels may be sparse. Our approach is significantly better than all baselines that can operate on all resolutions. Note that, the insertion metric is higher at lower resolutions, because a larger chunk of image with more complete context information is inserted at every point. Hence, a few percentage points lower insertion metric at higher resolutions do not necessarily mean the

Table 3: Comparative evaluation in terms of runtime (averaged on 5,000 images) on the ImageNet dataset using ResNet50 as the base model.

Running time (s)	224×224	112×112	28×28	14×14	7×7
Mask	17.03	14.61	14.66	14.35	14.24
GradCam	--	--	--	--	<1
RISE	61.77	--	--	--	--
Integrated Gradients	<1	--	--	--	--
I-GOS (ours)	6.07	5.73	5.70	5.63	5.62

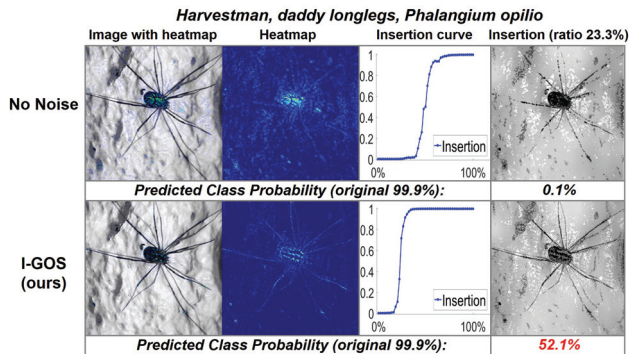


Figure 6: Examples from ablation studies (at 224×224 resolution). With added noise, the heatmap successfully reveals the entire legs of *daddy longlegs*, leading to better insertion metric, whereas without noise it is more adversarial (maybe merely by breaking each leg, CNN confidence is already reduced), leading to worse insertion metric

heatmaps are any worse. In practice, 28×28 heatmaps are usually more visually appealing, but in order to capture thin parts, we sometimes need to resort to 224×224 (e.g. Fig. 6).

Speed: In Table 3, we summarize the average runtime for Mask, RISE, GradCam, Integrated Gradients, and I-GOS on the ImageNet dataset using ResNet50 as the base model. For each approach, we only use one Nvidia 1080Ti GPU. For I-GOS, the maximal iteration is 15; for Mask, the maximal iteration is 500. Our approach is faster than Mask and RISE. Especially, it converges quickly, with the average number of iterations to converge being 13 and the time for each iteration being 0.38s. The average running times for the backpropagation-based methods are all less than 1 second. However, our approach achieve much better performance than these approaches, especially with higher resolutions. To the best of our knowledge, our approach I-GOS is the fastest among the perturbation-based methods, as well as the one with the best performance in deletion and insertion metrics among all heatmap approaches.

Table 4: The results of the ablation study on VGG19.

	224×224		28×28	
	Deletion	Insertion	Deletion	Insertion
I-GOS				
Ours	0.0336	0.5246	0.0899	0.5701
No TV term	0.0308	0.3712	0.0841	0.5181
No noise	0.0559	0.4194	0.0872	0.5634
Fixed step size	0.0393	0.5024	0.0906	0.5403

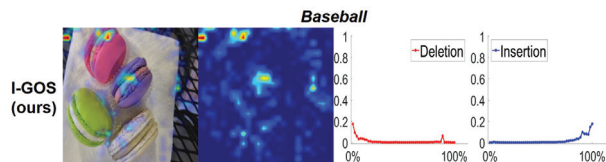


Figure 7: One failure case for I-GOS, insertion curve does not move until almost all pixels have been inserted.

Table 5: The optimization loss on VGG19 for resolution 28×28 .

	$\lambda_1 = 0.01, \lambda_2 = 0.2$		$\lambda_1 = 0.1, \lambda_2 = 2$		$\lambda_1 = 1, \lambda_2 = 20$	
	I-GOS	Mask	I-GOS	Mask	I-GOS	Mask
Total loss	0.2241	0.3349	0.3739	0.4857	0.6098	0.6794
Deletion	0.0825	0.1056	0.0861	0.1178	0.0899	0.1340
Insertion	0.5418	0.5335	0.5624	0.5307	0.5701	0.5207

Ablation Studies: We show the results of ablation studies in Table 4. From Table 4 we observe that without the TV term, insertion scores would indeed suffer significantly while deletion scores do not change much, indicating that the TV term is important to avoid adversarial masks. The random noise introduced in section *Avoiding adversarial examples* of the paper is very useful when the resolution of the mask is high (e.g. 224×224). From Fig. 6 we observe that I-GOS with noise can achieve much better insertion scores than without noise for the same insertion ratio. When the resolution is low (e.g. 28×28), the noise is not that important since low resolution can already avoid adversarial examples. When we utilize a fixed step size (the step size is 1 in Table 4), both deletion and insertion scores become worse, showing the utility of the line search.

Failure Case: Fig. 7 shows one failure case, where I-GOS found an adversarial mask and the insertion score did not increase till the end. Our observation is that optimization-based methods such as I-GOS usually do not work well when the deep model’s prediction confidence is very low (less than 0.01), or when the deep model makes a wrong prediction.

Convergence: For the values of the objective after convergence with Mask (Fong and Vedaldi 2017) vs. the proposed I-GOS, Table 5 shows the comparison at 28×28 with different parameters. Best parameters were used for each approach in Table 1 (0.01/0.2 for Mask and 1/20 for I-GOS). It can be seen at every parameter setting I-GOS has lower total loss than Mask (total loss is higher with larger λ_1 and λ_2 since the L1+TV terms have higher weights in total loss).

Conclusion

In this paper, we propose a novel visualization approach I-GOS, which utilizes integrated gradients to optimize for a heatmap. We show that the integrated gradients provides a better direction than the gradient when a good baseline is known for part of the objective of the optimization. The heatmaps generated by the proposed approach are human-understandable and more correlated to the decision-making

of the model. Extensive experiments are conducted on three benchmark datasets with four pretrained deep neural networks, which shows that I-GOS advances state-of-the-art deletion and insertion scores on all heatmap resolutions.

Acknowledgments

This work was partially supported by DARPA contract N66001-17-2-4030.

References

- Alvarez-Melis, D., and Jaakkola, T. S. 2018. Towards robust interpretability with self-explaining neural networks. In *NIPS*, 7786–7795.
- Ancona, M.; Ceolini, E.; Öztireli, A. C.; and Gross, M. H. 2017. A unified view of gradient-based attribution methods for deep neural networks. *CoRR* abs/1711.06104.
- Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.; and Samek, W. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS One* 10.
- Bau, D.; Zhou, B.; Khosla, A.; Oliva, A.; and Torralba, A. 2017. Network dissection: Quantifying interpretability of deep visual representations. In *Computer Vision and Pattern Recognition*.
- Dabkowski, P., and Gal, Y. 2017. Real time image saliency for black box classifiers. In *NIPS*.
- Elenberg, E.; Dimakis, A. G.; Feldman, M.; and Karbasi, A. 2017. Streaming weak submodularity: Interpreting neural networks on the fly. In *Advances in Neural Information Processing Systems*, 4044–4054.
- Fong, R. C., and Vedaldi, A. 2017. Interpretable explanations of black boxes by meaningful perturbation. In *2017 IEEE International Conference on Computer Vision (ICCV)*, 3449–3457.
- Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In Ghahramani, Z.; Welling, M.; Cortes, C.; Lawrence, N. D.; and Weinberger, K. Q., eds., *Advances in Neural Information Processing Systems 27*. Curran Associates, Inc. 2672–2680.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Johnson, S., and Subha, T. 2017. A study on eye fixation prediction and salient object detection in supervised saliency. *Materials Today: Proceedings* 4(2, Part B):4169 – 4181. International Conference on Computing, Communication, Nanophotonics, Nanoscience, Nanomaterials and Nanotechnology.
- Kruthiventi, S. S. S.; Gudisa, V.; Dholakiya, J. H.; and Venkatesh Babu, R. 2016. Saliency unified: A deep architecture for simultaneous eye fixation prediction and salient object segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Kummerer, M.; Wallis, T. S. A.; Gatys, L. A.; and Bethge, M. 2017. Understanding low- and high-level contributions to fixation prediction. In *The IEEE International Conference on Computer Vision (ICCV)*.
- Lundberg, S. M., and Lee, S.-I. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, 4765–4774.
- Nie, W.; Zhang, Y.; and Patel, A. 2018. A Theoretical Explanation for Perplexing Behaviors of Backpropagation-based Visualizations. *ArXiv e-prints*.
- Nocedal, J., and Wright, S. 2000. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer New York.
- Petsiuk, V.; Das, A.; and Saenko, K. 2018. RISE: Randomized Input Sampling for Explanation of Black-box Models. *ArXiv e-prints*.
- Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. 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*, 1135–1144. ACM.
- Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; Berg, A. C.; and Fei-Fei, L. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*.
- Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, 618–626.
- Shrikumar, A.; Greenside, P.; Shcherbina, A.; and Kundaje, A. 2016. Not just a black box: Learning important features through propagating activation differences. *CoRR* abs/1605.01713.
- Simonyan, K., and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations*.
- Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2014. Deep inside convolutional networks: Visualising image classification models and saliency maps. *ICLR Workshop*.
- Springenberg, J.; Dosovitskiy, A.; Brox, T.; and Riedmiller, M. 2015. Striving for simplicity: The all convolutional net. In *ICLR Workshop*.
- Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. In Precup, D., and Teh, Y. W., eds., *Proceedings of the 34th International Conference on Machine Learning*, 3319–3328. PMLR.
- Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2014. Intriguing properties of neural networks. In *International Conference on Learning Representations*.
- Zeiler, M. D., and Fergus, R. 2014. Visualizing and understanding convolutional networks. In Fleet, D.; Pajdla, T.; Schiele, B.; and Tuytelaars, T., eds., *Computer Vision – ECCV 2014*, 818–833. Cham: Springer International Publishing.
- Zhang, J.; Lin, Z.; Brandt, J.; Shen, X.; and Sclaroff, S. 2016. Top-down neural attention by excitation backprop. In *European Conference on Computer Vision*, 543–559. Springer.
- Zhang, Q.; Wu, Y. N.; and Zhu, S. 2018. Interpretable convolutional neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 8827–8836.
- Zhou, B.; Khosla, A.; Lapedriza, À.; Oliva, A.; and Torralba, A. 2014. Object detectors emerge in deep scene cnns. *CoRR* abs/1412.6856.
- Zhou, B.; Sun, Y.; Bau, D.; and Torralba, A. 2018. Interpretable basis decomposition for visual explanation. In *The European Conference on Computer Vision (ECCV)*.