arXiv:2305.20052v1 [cs.LG] 31 May 2023

Integrated Decision Gradients: Compute Your Attributions Where the Model Makes Its Decision

Chase Walker
Department of ECE
University of Central Florida
Orlando, FL, USA
chase.walker@knights.ucf.edu

Sumit Jha
Computer Science Department
University of Texas at San Antonio
San Antonio, TX, USA
sumit.jha@utsa.edu

Kenny Chen
Lockheed Martin Corporation
Orlando, FL, USA
kenny.chen@lmco.com

Rickard Ewetz
Department of ECE
University of Central Florida
Orlando, FL, USA
rickard.ewetz@ucf.edu

Abstract

Attribution algorithms are frequently employed to explain the decisions of neural network models. Integrated Gradients (IG) is an influential attribution method due to its strong axiomatic foundation. The algorithm is based on integrating the gradients along a path from a reference image to the input image. Unfortunately, it can be observed that gradients computed from regions where the output logit changes minimally along the path provide poor explanations for the model decision, which is called the *saturation effect* problem. In this paper, we propose an attribution algorithm called integrated decision gradients (IDG). The algorithm focuses on integrating gradients from the region of the path where the model makes its decision, i.e., the portion of the path where the output logit rapidly transitions from zero to its final value. This is practically realized by scaling each gradient by the derivative of the output logit with respect to the path. The algorithm thereby provides a principled solution to the saturation problem. Additionally, we minimize the errors within the Riemann sum approximation of the path integral by utilizing non-uniform subdivisions determined by adaptive sampling. In the evaluation on ImageNet, it is demonstrated that IDG outperforms IG, left-IG, guided IG, and adversarial gradient integration both qualitatively and quantitatively using standard insertion and deletion metrics across three common models.

1 Introduction

The access to internet-scale data and compute power has fueled the success of black box neural network models for applications such as disease detection [1], image synthesis [2], and protein folding [3]. The phenomenal performance of these networks comes from the large number of parameters and non-linear interactions among them. The complex and high dimensional dynamics makes it difficult to understand and visualize why a neural network makes a particular decision. To establish trustworthiness in neural network models, noteworthy research efforts have been devoted to interpretability and explainability [4]. Attribution methods provide model explanation by computing the contribution of each input feature to a model decision. Attribution methods broadly fall into perturbation based methods [5, 6], backpropagation based methods [7, 8], and gradient based methods [9, 10]. Gradient based methods are promising due to their strong axiomatic foundation, and model-agnostic implementation [10].

Preprint. Under review.

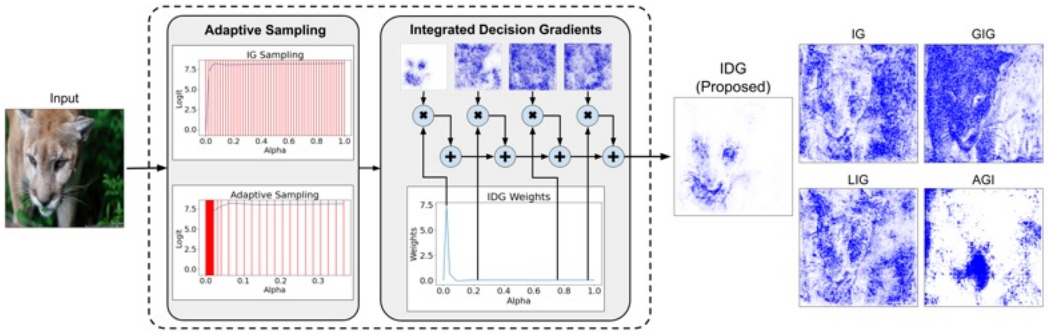


Figure 1: (left) An overview of the adaptive sampling algorithm, and the IDG attribution method. (right) A preliminary visual comparison of IDG with IG [10], LIG [11], GIG [12], and AGI [13].

Gradient based methods compute attribution maps by capturing the gradients at the model inputs with respect to the model outputs [9]. However, gradients computed with respect to important input pixels may be zero due to the non-linear activation functions. Integrated Gradients (IG) solved this problem by integrating the gradients along a path from a baseline reference image to the input image. Unfortunately, it can be observed that gradients from regions of the path where the output logit changes minimally (e.g. is saturated) provide poor explanations for the model decision [11]. This phenomena is called the *saturation effect* problem. Solution templates to solve the saturation problem include: selecting non-straight-line paths [12, 13], path truncation [11], and post processing methods that use thresholding [14] as well as averaging across blurred inputs [15]. While these methods improve attribution quality, they do not provide a principled solution to the saturation problem.

In this paper, we propose a new attribution method called Integrated Decision Gradients (IDG). We call the portion of the path where the output logit rapidly transitions from zero to its final value the *decision region*. The IDG algorithm focuses on integrating gradients from the decision region of the path integral. This is practically realized by scaling each gradient by the derivative of the output logit with respect to the path. The scaling factor rewards gradients in the decision region and penalizes gradients from outside the decision region. The main contributions of this paper are summarized as follows:

- We propose a new attribution method called IDG that satisfies a path integral sensitivity axiom and provides a principled solution to the saturation problem.
- We present an adaptive sampling technique to select non-uniform subdivisions for the Riemann approximation of the path integral. The non-uniform subdivisions reduce computational errors (or runtime overheads) compared with using uniform subdivisions.
- Compared with IG [10], Left-IG (LIG) [11], Guided IG (GIG) [12], and Adversarial Gradient Integration (AGI) [13], IDG improves both the qualitative and quantitative results.

The remainder of the paper is organized as follows: related work is examined in Section 2, the IDG attribution method in Section 3, the adaptive sampling algorithm is proposed in Section 4, experimental evaluation is presented in Section 5, and the paper is concluded in Section 6.

2 Related Work

In this section, we first review the limitations of directly using gradients as attributions. Next, we review integrated gradients and assess the saturation effect problem within path integrals.

2.1 Limitations of Using Gradients as Attributions

Attributions are defined to be the contribution of each input feature to the model output decision. An attribution method satisfies the axiom of *sensitivity* if a single feature that differs between a baseline and input - which produce different output predictions - is given a non-zero attribution. Additionally, if a neural network is not affected by changing a variable, then that variable's attribution shall be zero [10]. Computing the gradient of the inputs with respect to the output logit is a promising

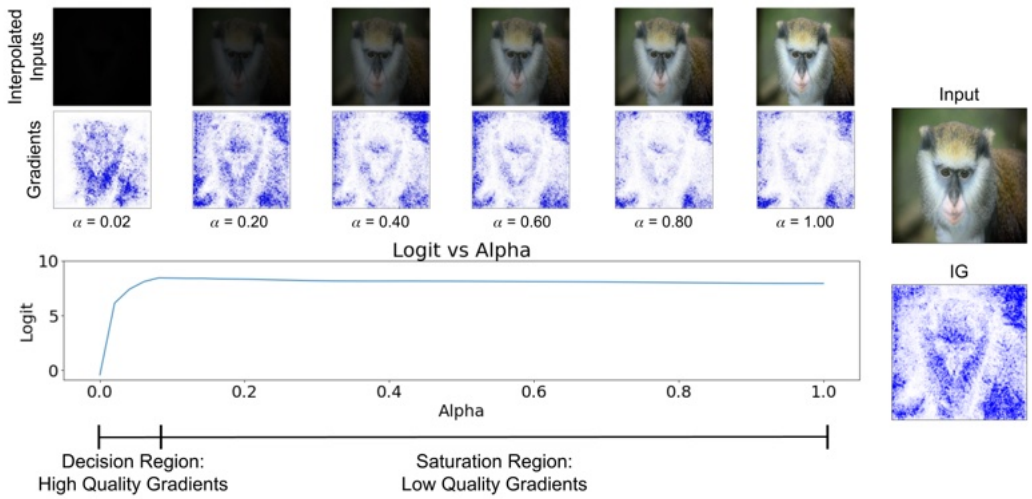


Figure 2: The figure illustrates the IG attribution method and saturation effects within path integrals. The top row shows interpolated inputs and the second row shows the corresponding gradients. The IG attribution map (shown to the right) is the average of the gradients. The third row shows the logit-α curve, which defines the decision and saturation regions. It can be observed that the gradients from the decision region are of higher quality than the saturation region.

method for computing attributions [9]. However, the use of non-linear activation functions causes the sensitivity axiom to be violated [10], which is shown in Example 1 below.

Example 1. *Consider a function $F = 1 - ReLU(1 - x)$, a baseline $x' = 0$, and an input $x = 2$. For $x' = 0$, the function F is equal to 0, and for $x = 2$, the function F is equal to 1. Since changing x from 0 to 2 affects the output of F, the attribution w.r.t. the feature x should be non-zero. However, $\partial F/\partial x = 0$ at $x = 2$, which results in an attribution of 0 [10].*

Integrated gradients offers a solution to computing attributions that satisfies the sensitivity axiom.

2.2 Integrated Gradients

Integrated Gradients computes attributions by integrating gradients on a straight line between a reference image and an input image [10]. Let F be the function realizing the output logit of interest. IG_i with input image x is mathematically defined using a path-integral [10], as follows:

$$IG_i(x) = (x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial F(x_i' + \alpha \times (x_i - x_i'))}{\partial x_i} d\alpha \tag{1}$$

where x' is a black baseline image, $\alpha \in [0, 1]$ parameterizes the straight-line path between x' and x, x_i and x_i' represent a single pixel of their respective images, and IG_i is therefore the attribution of pixel i in the input image.

The IG attribution method is illustrated in Figure 2. The top row shows interpolated inputs, the second row shows the corresponding input gradients, the third row visualizes the output logit with respect to the path. The IG attribution map is equal to the sum of the gradients in the second row. The use of a path-integral ensures that gradients from regions of F where $\partial F/\partial x_i$ is non-zero are computed. In Example 1, IG will compute gradients from the region $[0, 1]$, where $\partial F/\partial x = 1$. The resulting attribution w.r.t. x is 2, i.e., the attribution is non-zero and sensitivity is satisfied. Nevertheless, many attributions computed using IG are still noisy due to saturation effects [11].

2.3 Saturation Effects within Path-Integrals

To introduce and understand the *saturation effect* problem within path-integrals, we examine the performance of the IG attribution method in in Figure 2. We study the quality of the computed gradients with respect to the decision and saturated regions of the path integral. It can be observed

that (i) gradients from the saturation regions are of low quality and (ii) gradients from the decision region are of high quality. The conclusion is rather straight forward to understand. If the model output does not increase while moving $\triangle\alpha$ along the path, it is intuitive that the corresponding gradients are not important to the model decision. Conversely, if the output logit changes rapidly while moving $\triangle\alpha$ along the path, those gradients have a strong impact on the model decision.

This raises the rudimentary question: Is it possible to design a path integral that focuses on computing gradients from the region where the model decision is made and the highly informative gradients are located? It can for example be observed in Figure 2 that the gradients computed at $\alpha = 0.02$ alone provide an excellent explanation for the model decision.

3 Integrated Decision Gradients

In this section, we propose a new attribution method called Integrated Decision Gradients (IDG). We outline the motivation behind the design of IDG, explain the concept of importance factors, and provide the definition as well as a visualization of IDG.

3.1 Motivation

Path integrals integrate gradients from a reference image to an input target image. A fundamental challenge is to determine the ideal importance of each gradient. Based on the analysis in the previous section, we define a new *sensitivity axiom* for path integrals. Next, we introduce the concept of an importance factor, which is used to construct an attribution algorithm that satisfies the axiom.

Axiom: Sensitivity (path integrals) Let F be the output of a neural network. For every point within a path integral parameterized by a parameter α, an attribution method satisfies Sensitivity (path integrals) if there is no contribution to the attribution result when $\partial F/\partial\alpha$ is equal to zero. If $\partial F/\partial\alpha$ is non-zero, there should be a non-zero contribution to the attribution result.

None of the existing attribution methods based on path integrals satisfy the axiom [10, 11, 12, 13]. The traditional IG method places an equal weight on all gradients [10], even those that occur in the saturation region where $\partial F/\partial\alpha = 0$. The Left-IG attribution attempts to solve this by truncating the path integral after the output logit has reached 90% of its final value [11]. This assigns a weight of zero and one to gradients from the approximate saturation and decision regions respectively, which does not guarantee that the axiom is satisfied. GIG and AGI use non-straight line paths that attempt to avoid integrating gradients from saturated regions [12, 13], which does also not guarantee that the Sensitivity (path integrals) axiom is satisfied.

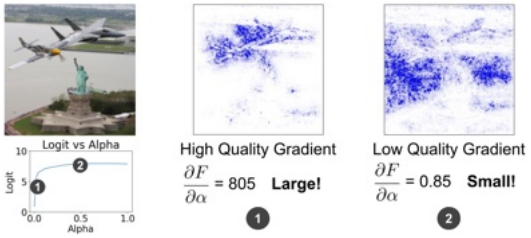


Figure 3: This figure illustrates the relationship between importance factor magnitude and gradient quality. Higher importance factors are directly related to higher quality gradients.

To satisfy the axiom, we conjecture that the importance of each gradient should be proportional to the impact on the model output, which is conceptually shown in Figure 3. Inspired by this, we define an *importance factor*, as follows:

$$IF(\alpha) = \frac{\partial F(x' + \alpha(x - x'))}{\partial \alpha} \tag{2}$$

where $IF(\alpha)$ is the importance of the gradient computed at α. Next, we define an attribution method that satisfies the Sensitivity (path integrals) axiom based on scaling each gradient with the importance factor in Eq (2).

3.2 Definition of Integrated Decision Gradients

In this subsection, we formally define the IDG attribution algorithm. Given a neural network represented by function $F : R^n \rightarrow [0, 1]$, an input vector x, and given F exists over the range

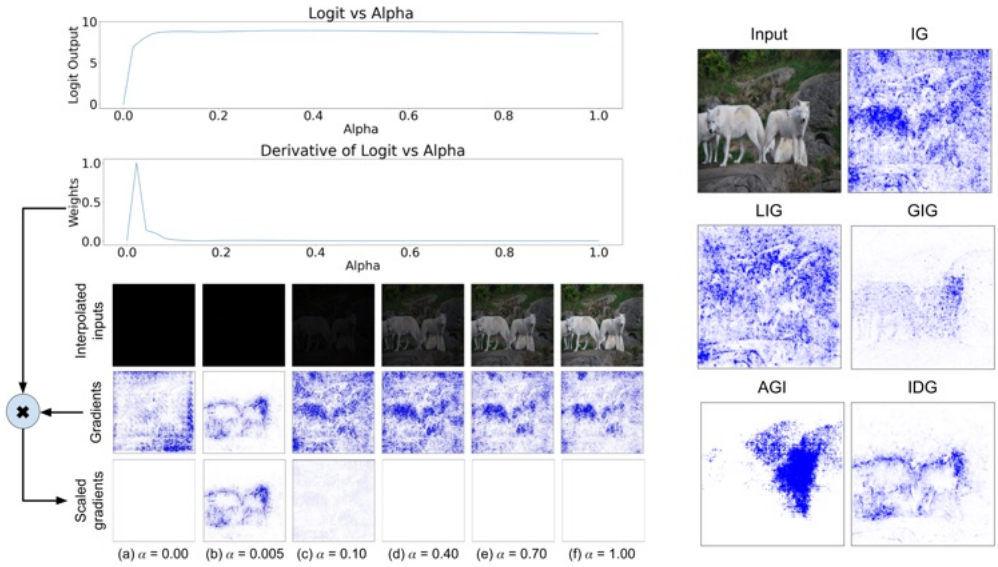


Figure 4: A full visualization of how IDG uses importance factors to eliminate saturation effects. The top row shows the logit-α curve. The next row shows the derivative of the curve, i.e., the importance factors with respect to α. The third row shows the interpolated images, the fourth shows the associated gradients, and the bottom row shows these gradients scaled with the corresponding importance factors. The right side shows the input image, and the attributions computed using IG [10], LIG [11], GIG [12], AGI [13], and IDG.

$\alpha \in [0, 1]$, IDG assigns an importance factor to each input feature x_i with respect to the model output, using the following equation:

$$IDG_i(x) = \underbrace{(x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial F(x_i' + \alpha(x_i - x_i'))}{\partial x_i}}_{\text{Traditional IG}} \times \underbrace{\frac{\partial F(x_i' + \alpha(x_i - x_i'))}{\partial \alpha}}_{\text{Importance Factor}} d\alpha \tag{3}$$

The IDG attribution method is equivalent to IG in Eq (1) but with each gradient scaled with the importance factor in Eq (2). The importance factor is equivalent to the derivative of the logit-α curve in the bottom of Figure 2. The importance factors scale-up gradients from the decision region and scale-down gradients from saturated regions, respectively. Therefore, IDG provides a principled solution to the saturation problem, and satisfies the Sensitivity (path integrals) axiom by definition.

The path integral is practically computed using the Riemann sum approximation [10], as follows:

$$IDG_i(x) = (x_i - x_i') \times \frac{1}{m} \times \sum_{k=1}^{m} \frac{\partial F(x_i' + \frac{k}{m} \times (x_i - x_i'))}{\partial x_i} \times \frac{\partial F(x_i' + \frac{k}{m} \times (x_i - x_i'))}{\partial \alpha} \tag{4}$$

where m is the number of steps for approximation. We will further discuss the selection of the step size and its impact on the approximation error in Section 4.

We illustrate IDG with an example in Figure 4. First, looking at the left side of the figure, the top row shows the logit-α curve associated with the input image. The second row shows the derivative of this curve, i.e., $\partial F / \partial \alpha$ in Eq (2). The third row shows the interpolated inputs for selected alpha values and the fourth row shows the gradients computed by IG for these inputs. The last row visualizes the effect of IDG by scaling the gradients above by the importance factors from the second graph. The importance factors scale up the magnitude of the gradients from the decision region while scaling down the magnitude of the gradients from the saturated regions. In the figure, it can be observed that, in particular, the attributions from $\alpha = 0.005$ are scaled up. On the right of the figure, we show the original image, and the attributions generated by IG, LIG, GIG, AGI, and IDG. The attributions computed using IDG are substantially less noisy than all competitors. We note that GIG has a low amount of noise, but IDG has more focused attributions on the highlighted features.

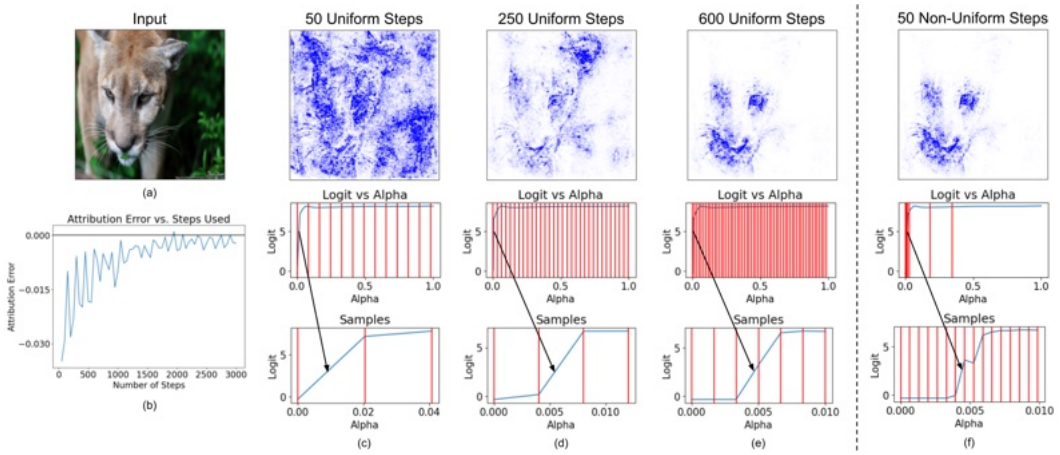


Figure 5: This figure shows the motivation for the adaptive sampling algorithm. The image (a) is the input to the attributions in the figure. The graph (b) demonstrates how the attribution error decreases as step count increases. Columns (c), (d), and (e) of attributions and graphs show the relationship between sample locations and IDG quality as 50, 250, and 600 steps are used respectively. We show that as the number of steps increases, the quality of IDG grows greatly, influencing the adaptive sampling algorithm. Lastly, column (f) shows the equivalent result of column (e) achieved by using adaptive sampling with 50 steps.

4 Adaptive Sampling Algorithm

In this section, we first analyze the errors within the Riemann sum approximation of the IDG path integral for uniform subdivisions. Next, we propose an adaptive sampling technique to minimize the approximation errors using non-uniform subdivisions. In the supplementary results, we show that the adaptive sampling only results in major improvements for IDG. The impact of the adaptive sampling on regular IG is minor.

4.1 Motivation

The errors within the Riemann approximation of the IDG path integral can be calculated, as follows:

$$\epsilon(x, n) = \lim_{m \to \infty} IDG_i(x, m) - IDG_i(x, n) \tag{5}$$

where $\epsilon(x_i, n)$ is the approximation error for attribution x_i when computing the integral with n uniform subdivisions. n and m are the number of steps used within the Riemann sum approximation in Eq (4).

We analyze the approximation error and the impact on the attributions in Figure 5. The graph (b) shows the average error across all the pixels in the attribution map with respect to the number of used steps n. Since a low step count results in a lack of samples in the decision region, a large number of steps are required for a good approximation. The image (a) is the input for the four columns (c), (d), (e), and (f) of attributions and graphs. The columns show the quality of the attributions with respect to the number of steps and type of subdivision. It is observed from the graphs that taking more samples in the decision region greatly improves IDG attribution quality. Therefore, to obtain high IDG quality without a prohibitive number of steps, we design a new adaptive sampling algorithm - seen in Figure 5 (f) - that uses non-uniform subdivisions concentrated on the decision region.

4.2 Adaptive Sampling Methodology

It is desirable to sample the high quality gradients that lie in the decision region to improve the quality of the attained attributions. In Algorithm 1, we show how the adaptive sampling algorithm is used with IDG. Our approach is based on first pre-characterizing the logit-α curve with N uniform subdivisions in lines 3 - 7. Next, M subdivisions are non-uniformly distributed within the N regions based on logit growth and IDG is calculated in lines 8 - 15. Because there are M total samples, line

Algorithm 1 Computing IDG with Adaptive Sampling

Input: Model F, image x, baseline x', pre-characterization steps N, number of IDG steps M
Output: An attribution map A

$x^0 = x'$
$x^{N-1} = x$
$samples[0] = 0$
// Pre-characterization of logit-α curve
for $i = 0$ to $N - 1$ **do**
  $x^{i+1} = x' + \frac{i}{N} \times (x - x')$
  $samples[i+1] = round(\frac{F(x^{i+1}) - F(x^i)}{F[x^{N-1}] - F[x^0]} \times M)$
end for
// Computation of IDG with non-uniform samples
for $i = 0$ to N **do**
  for $j = 0$ to $samples[i]$ **do**
    $\alpha = \frac{i}{N} + \frac{j}{N \times samples[i]}$
    $x^i = x' + \alpha \times (x - x')$
    $IDG[i] = \frac{\partial F(x^i)}{\partial x} \times \frac{\partial F(x^i)}{\partial \alpha} \times \frac{1}{N \times samples[i]}$
  end for
end for
return $A = mean(IDG)$

11 executes $O(N + M)$ times. In practice it is best if $N = M$ (this is shown in the supplementary materials) therefore the algorithm runtime is $O(N)$.

As seen in Figure 5 (e) and (f), combining this adaptive sampling algorithm with IDG creates attributions as strong as IDG with 600 steps while only using 50 steps. Figure 1 provides a high-level overview of this new IDG process. The figure shows that when given an input image and a number of steps, the adaptive sampling algorithm calculates non-uniform subdivisions based on logit growth. These are then used as input for IDG where the gradient at each location is calculated and then weighted, producing the final attribution. In this figure, the IDG sampling graph shows that 31 out of 50 samples are placed in the decision region $\alpha \in [0.0, 0.2]$, where the logit changes from 0 to 7.2.

5 Experimental Results

In this section, we will evaluate the effectiveness of the proposed method. We perform our experiments in PyTorch using the 2012 validation set of ImageNet [16] on NVIDIA A40 GPUs. According to ML CO_2 impact, the experimental evaluation released 43.6 kg of CO_2 with zero offset [17]. The attributions computed using Algorithm 1 are called IDG. We compare our method with IG [10], left-IG [11], guided IG [12], and adversarial gradient integration [13]. We use Captum for the implementation of IG, whereas left-IG, GIG, and AGI are taken from their respective repositories [18, 19, 20, 21]. We evaluate the quality of the computed attributions both quantitatively and qualitatively.

In Table 1, we quantitatively evaluate the attributions using standard perturbation testing which measures the importance of the pixels in an attribution via an area under the curve (AUC) score. A total of four tests are presented with three insertion methods and one deletion method from the authors of RISE and XRAI [22, 14] which are described in Section 5.1. The table compares the computed attribution quality for the first 5000 images of the ImageNet dataset such that five images are taken from each of the 1000 classes. The five attribution methods are evaluated with three models trained on ImageNet. We selected ResNet101 (R101), ResNet152 (R152), and ResNeXt (RNXT) as pre-trained models from PyTorch and use the newest ImageNet weights available (V2 for the ResNet models and V1 for ResNeXt) [23, 24, 25].

Qualitatively, we present a subset of five examples in Figure 6 which are gathered gathered using the ResNet101 model and the method parameters explained below. We provide a larger selection of examples in the supplementary materials for visual comparison.

Table 1: Comparison of attributions using the AIC, SIC, insertion, and deletion tests

Metric	Model	IG [10]	LIG [11]	GIG [12]	AGI [13]	IDG
AIC (↑)	R101	0.571	0.589	0.626	0.675	**0.701**
	R152	0.575	0.616	0.646	0.686	**0.718**
	RNXT	0.580	0.611	0.634	0.654	**0.730**
SIC (↑)	R101	0.498	0.522	0.559	0.609	**0.638**
	R152	0.508	0.552	0.582	0.619	**0.659**
	RNXT	0.478	0.518	0.532	0.554	**0.620**
Insertion (↑)	R101	0.498	0.535	0.547	0.561	**0.592**
	R152	0.517	0.562	0.565	0.577	**0.615**
	RNXT	0.276	0.299	0.296	0.307	**0.324**
Deletion (↓)	R101	0.181	0.148	0.155	0.172	**0.108**
	R152	0.202	0.148	0.164	0.190	**0.118**
	RNXT	0.101	0.078	0.082	0.104	**0.068**

Inputs are reshaped to $(224, 244)$ for all three presented models. This image processing follows the attribution documentation provided by Captum [18]. The RISE, AIC, and SIC tests use the default parameters found from their respective repositories [26, 27]. The IG and LIG attribution methods use 50 steps and a black baseline image. GIG uses the default parameters found at [20]. AGI uses the default parameters found at [21]. Lastly, IDG is used with 50 steps and a black baseline image. For all the methods, we use a single baseline only.

5.1 Quantitative Evaluation Metrics

The evaluation metrics are built upon the intuition that the highest attribution values should correspond to those features that contribute more to the classification of the target class [22, 14]. The process starts from the most important pixels and starts deleting (inserting) them from the original image (to a blurred image for insertion) until only a black (the original) image remains. At each step, the softmax score (or accuracy) is calculated. This gives us an ROC curve from base image to final image, which is used to compute the AUC score for a given attribution. This AUC value is computed for each image and then averaged out over the entire test data selection. For the insertion game, a higher AUC score indicates a better attribution and for the deletion game, a lower AUC score indicates better performance. The two sets of methods presented from Petsiuk, et al and Kapishnikov, et al. take different approaches to the insertion process [22, 14].

In RISE, the insertion (deletion) test which starts (ends) with a Gaussian blurred (black) image [22]. In their implementation pixels are added (deleted) in equal amounts during the test process. Given an NxN image, the test will change the image by N pixels at a time over N steps.

Kapishnikov, et al. present the Accuracy Information Curve (AIC) and Softmax Information Curve (SIC) in their XRAI paper [14]. The AIC test gives each perturbation step a score of 0 or 1 for an incorrect or correct classification and SIC uses softmax as previously discussed. For pixel perturbation, these methods use a schedule that non-linearly removes groups of pixels from the image in increasingly large amounts. The last difference from the RISE insertion test is the blurring method, where the initial image is now blurred in segments, each having its own noise distribution.

5.2 Comparison With Previous Work

In Table 1, attribution quality is evaluated using the AIC and SIC insertion metrics and the RISE insertion and deletion metrics. We use an arrow to denote if larger (arrow up) or smaller (arrow down) scores are better. The best score for each model and test type is in bold. Additionally we provide how many times a given method outperforms all other methods in the last row of the table.

It can be observed in Table 1 that IDG achieves a consistent improvement over IG, LIG, GIG, and AGI across all twelve of the tests presented. Comparing IDG to IG and LIG clearly indicates the ability of IDG to mitigate saturation effects in path-based methods while retaining the most important gradient information. When compared to AGI and GIG, the large margin of improvement in the scores shows

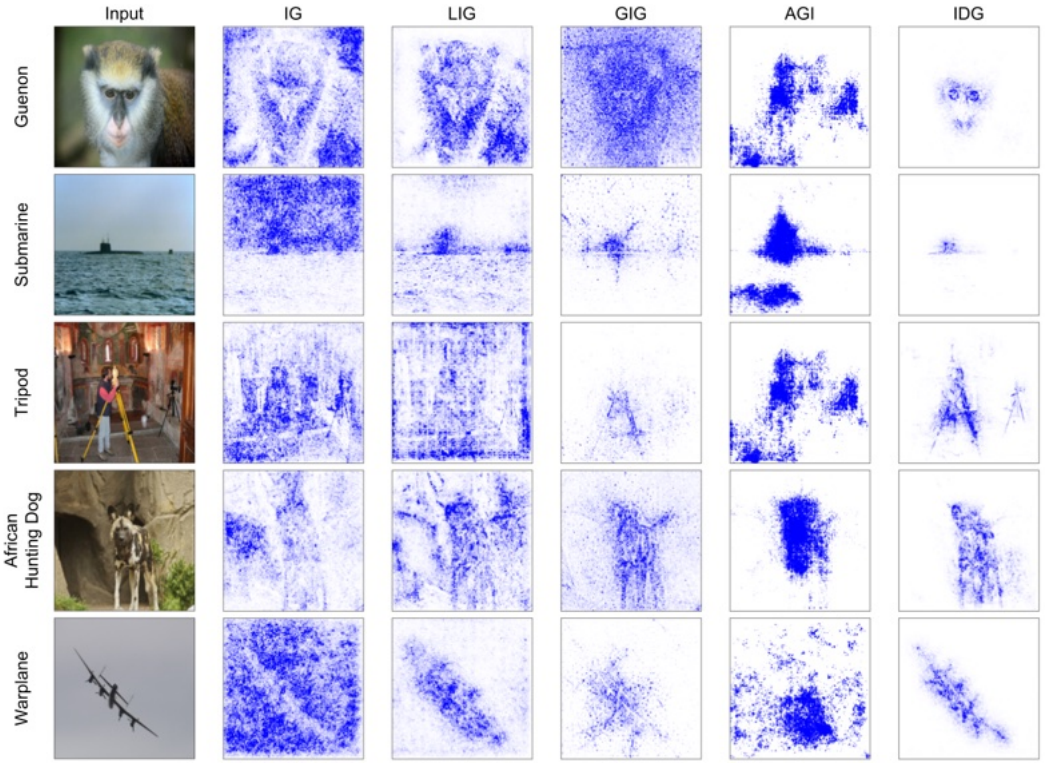


Figure 6: Qualitative comparison of attributions computed using the IG [10], LIG [11], GIG [12], and AGI [13], and IDG methods. It is seen that IDG solves the saturation problem and outperforms the state-of-the-art path-based attribution methods in visual quality.

that IDG presents a more complete solution to the saturation problem than these methods. Overall, IDG outperforms all of the attribution methods in the comparison, achieving new state-of-the-art performance.

For qualitative analysis, we compare IG, LIG, GIG, AGI, and IDG in Figure 6. All attributions are computed as previously described. The comparison is performed using images of a “Guenon", “Submarine", “Tripod", “African Hunting Dog", and “Warplane" taken from ImageNet [16].

Across the five selections, IDG clearly produces much sharper attributions than IG and LIG, further verifying that it solves the saturation problem present in these methods. When compared to GIG, IDG also has superior performance in all of the images. For the Tripod example, even though GIG has relatively low noise, IDG has stronger attributions on the tripod in the foreground, and the one in the background as well. Lastly, when comparing to AGI, it can be seen AGI generally has low extraneous noise in the attributions. However, IDG provides tighter, and sharper attributions on the class subject in the images, therefore the results are better.

The images clearly show that IDG improves visual quality over the other methods. IDG generates attributions with less random noise, and sharper attributions, showing its ability to solve the saturation problem. Additionally it shows its ability to outperform the methods which use non-straight-line paths. We provide an additional 50 visual comparisons in the supplementary results section.

6 Discussion

In this paper, we propose a new attribution method called Integrated Decision Gradients (IDG). The key idea of IDG is to perform the path integral while weighting sampled gradients by their associated logit growth. This amplifies gradients located in the decision region, and negates those from the saturation region, solving the saturation issue. In contrast, traditional IG integrates gradients between the same images while giving all gradients equal weight, saturated or not, causing the majority of saturated gradients to dominate the output. Additionally, we provide evidence that

the decision region of the path integral is where the best gradients lie. With this, we present an adaptive sampling algorithm which densely samples the decision region without runtime penalty, improving IDG performance. We show qualitatively and quantitatively that IDG reaches state-of-the-art performance in the path-based attribution field. In our future work, we plan to apply IDG concepts to other attribution methods to further enhance attribution quality. We also plan to employ IDG within practical real-world applications. The code to replicate the results presented in this paper is available at: `https://github.com/chasewalker26/Integrated-Decision-Gradients`.

Limitations We present quantitative and qualitative results that show the proposed method outperforms those in its field. However, there does not yet exist criteria to perfectly examine what makes a good attribution. Therefore, we provide the best, currently accepted evaluation of our method.

Acknowledgements

This work was partly supported by the Lockheed Martin University Engagement Program, the Florida High Tech Corridor Matching Grants Program, and the DARPA cooperative agreement #HR00112020002. The views, opinions and/or findings expressed are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

References

[1] Meherwar Fatima, Maruf Pasha, et al. Survey of machine learning algorithms for disease diagnostic. *Journal of Intelligent Learning Systems and Applications*, 9(01):1, 2017.

[2] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[3] Milot Mirdita, Konstantin Schütze, Yoshitaka Moriwaki, Lim Heo, Sergey Ovchinnikov, and Martin Steinegger. Colabfold: making protein folding accessible to all. *Nature Methods*, pages 1–4, 2022.

[4] Arun Das and Paul Rad. Opportunities and challenges in explainable artificial intelligence (xai): A survey. *arXiv preprint arXiv:2006.11371*, 2020.

[5] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[6] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "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*, pages 1135–1144, 2016.

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

[8] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[9] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps, 2013.

[10] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3319–3328. JMLR.org, 2017.

[11] Vivek Miglani, Narine Kokhlikyan, Bilal Alsallakh, Miguel Martin, and Orion Reblitz-Richardson. Investigating saturation effects in integrated gradients, 2020.

[12] A. Kapishnikov, S. Venugopalan, B. Avci, B. Wedin, M. Terry, and T. Bolukbasi. Guided integrated gradients: an adaptive path method for removing noise. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5048–5056, Los Alamitos, CA, USA, jun 2021. IEEE Computer Society.

[13] Deng Pan, Xin Li, and Dongxiao Zhu. Explaining deep neural network models with adversarial gradient integration. In Zhi-Hua Zhou, editor, *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 2876–2883. International Joint Conferences on Artificial Intelligence Organization, 8 2021. Main Track.

[14] A. Kapishnikov, T. Bolukbasi, F. Viegas, and M. Terry. Xrai: Better attributions through regions. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4947–4956, Los Alamitos, CA, USA, nov 2019. IEEE Computer Society.

[15] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise, 2017.

[16] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[17] Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*, 2019.

[18] Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, et al. Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*, 2020.

[19] Vivek Miglani, Narine Kokhlikyan, Bilal Alsallakh, Miguel Martin, and Orion LLC Reblitz-Richardson. *Left-IG Code Repository*, 2020. Available at `https://github.com/vivekmig/captum-1/tree/ExpandedIG`.

[20] A. Kapishnikov, S. Venugopalan, B. Avci, B. Wedin, M. Terry, and T. Bolukbasi. Gig code repository, 2021. Available at `https://github.com/PAIR-code/saliency/tree/master/saliency/core`.

[21] Deng Pan, Xin Li, and Dongxiao Zhu. Agi code repository, 2021. Available at `https://github.com/pd90506/AGI`.

[22] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[24] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[25] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Curran Associates Inc., 2019.

[26] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise code repository, 2018. Available at `https://github.com/eclique/RISE`.

[27] A. Kapishnikov, S. Venugopalan, B. Avci, B. Wedin, M. Terry, and T. Bolukbasi. Aic code repository, 2019. Available at `https://github.com/PAIR-code/saliency/tree/master/saliency/metrics`.

[28] Content European Commission, Directorate-General for Communications Networks and Technology. Ethics guidelines for trustworthy ai, 2019.

[29] Mark Ryan. In ai we trust: ethics, artificial intelligence, and reliability. *Science and Engineering Ethics*, 26(5):2749–2767, 2020.

A Appendix

In this appendix we provide additional information that did not fit in the bounds of the paper. In Section A.1 we provide detailed analysis of the selection of N and M for the adaptive sampling algorithm. In Section A.2 we provide further explanation of the impact of the AS algorithm, showing that IDG provides the true solution to the saturation problem. In Sections A.3 and A.4 we provide information on the licensing of libraries used in experimental evaluation and we discuss the potential ethical impact of the proposed method. Lastly, in Section A.5 we present 50 additional qualitative visual comparisons of our proposed method against those presented in the manuscript.

A.1 Ablation Study for Adaptive Sampling

The adaptive sampling algorithm has two parameters N and M. N is the number of samples used in the pre-characterization of the logit-α curve. M is the number of samples used in the computation of IDG using non-uniform subdivisions. Three types of selections of N and M are possible: $N < M$, $N == M$, and $N > M$. Assuming M or N is set to 50, which is a common step count for path-based methods, we provide analysis of which selection provides the best result via an ablation study.

In Figure 7, we present an ablation study on the selection of N and M. In (a), M is set to 50 and we take the average of the deletion score [22] over 10 images as N is varied from 5 to 100 by increments of 5. In (b), N is set to 50 and the deletion scores are gathered as before where M is varied instead. We see from graph (a) that low values of N produce poor results and the transition from 5 to 20 results in a large drop in deletion score. We see a similar case in (b) where the score improves as M increases. We note that stable performance is seen on both graphs where $N == M$.

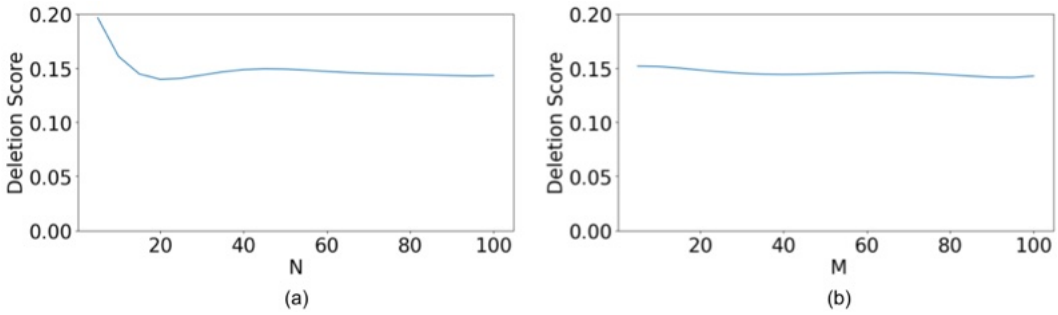


Figure 7: The change in the deletion score of IDG with AS averaged over 10 images by varying (a) N and (b) M. In the graphs, N (M) is varied from 5 to 100 while M (N) is set to 50. It is seen that the most stable scores are located where $N == M$.

We conclude that selecting $N == M$ results in proper estimation of the importance factors given the M IDG steps available for placement. While selecting $N < M$ may provide equally strong results, it may provide poor results without meaningful runtime improvement, therefore $N == M$ is chosen. Additionally, we note selecting an $N > M$ does not improve the score enough for the associated runtime penalty.

A.2 IDG Is the Solution to the Saturation Problem

In the manuscript we present adaptive sampling as a method to improve the ability of the proposed IDG method to solve the saturation problem. Adaptive sampling takes advantage of the importance factors to perform non-uniform sampling that focuses on the region of growth. This provides a large ratio of high quality gradients to saturated gradients from which IDG can generate an attribution. However, AS alone is not a solution to the saturation problem, which we will demonstrate by evaluating IG with adaptive sampling.

In Figure 8, for the given input image we compare the attributions generated by IG with uniform sampling (US), IG with adaptive sampling, IDG with uniform sampling, and IDG with adaptive sampling. When comparing IG with US and IG with AS, we see a small reduction in noise in the AS attribution, as there are inherently less saturated gradients captured when AS is applied to IG. However, due to IG equally weighting all gradients, the saturated gradients still dominate the output,

illustrating that AS alone cannot solve the saturation problem. However, when viewing IDG with US compared to both IG attributions, we see a vast improvement to attribution quality, illustrating IDG's ability to solve the saturation problem. Furthermore, when AS is is applied to IDG, its ability to solve the saturation problem increases.

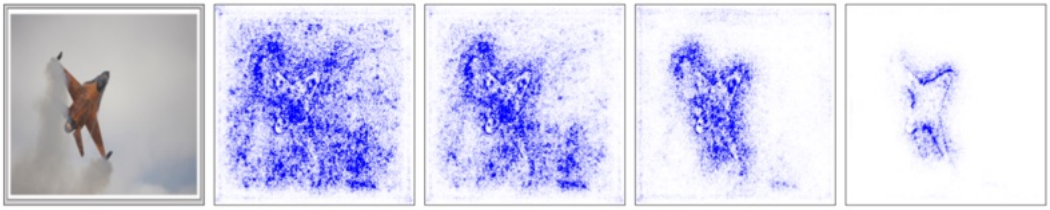


Figure 8: Given the input, attributions created by IG using uniform sampling (US) and AS are compared to attributions created by IDG with US and AS. Since AS applied to IG does not meaningfully improve performance over IG with US, and IDG with US provides a higher quality attribution, we determine that IDG, not AS, is the solution to the saturation problem. This is further exemplified by the improvement seen in IDG with AS, reinforcing the idea that AS gives IDG access to better gradients, but IDG is the solution to the saturation problem.

As adaptive sampling does not meaningfully improve IG performance and IDG with US provides much stronger attributions than IG, we verify that integrated decision gradients is the solution to the saturation problem. We reiterate that adaptive sampling is used to provide IDG access to gradients of a higher quality than US does, therefore improving its performance, but not acting as the solution to the saturation problem.

A.3 Licenses of Use

The 2012 validation set of ImageNet [16] is used under the BSD 3-Clause License. The insertion, deletion, AIC, and SIC are used under the MIT and Apache 2.0 licenses respectively [22, 14]. The IG [10] and LIG [11] attribution methods as well as the PyTorch repository [25] are used with the BSD 3-Clause license, GIG [12] is used under Apache License 2.0, and AGI [13] is used under the MIT license.

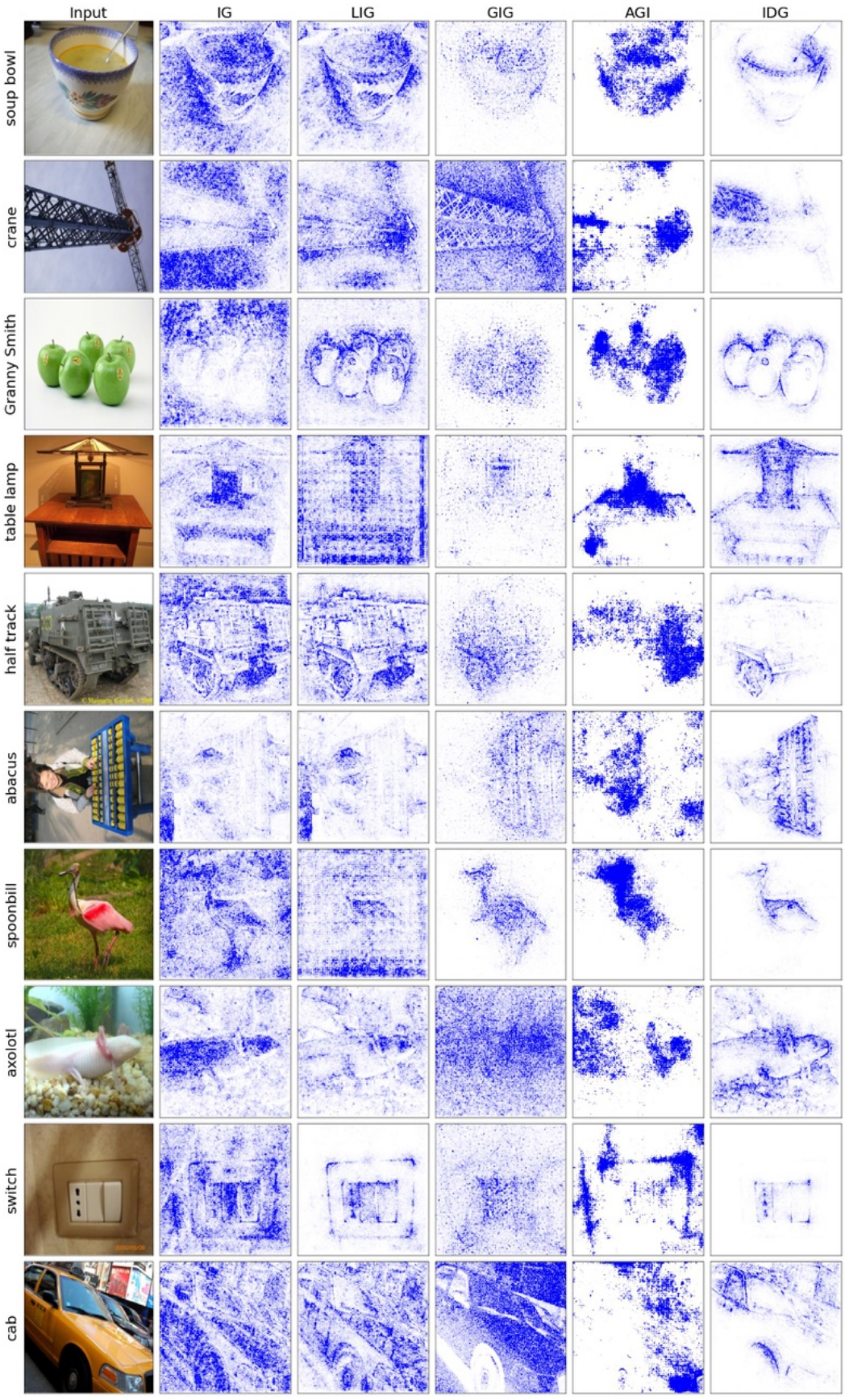
A.4 Broader Impact

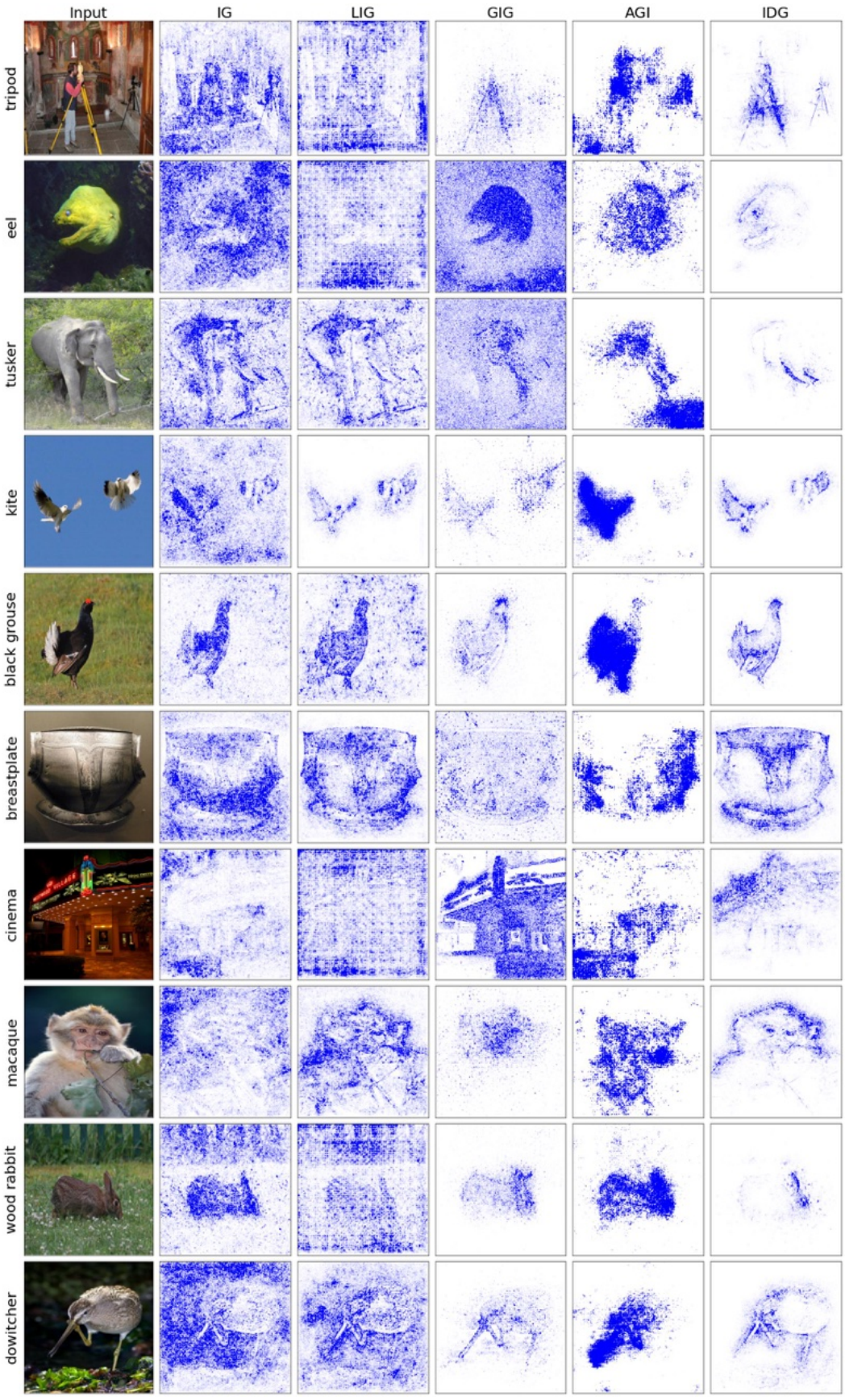
While explainable AI endeavours to increase human trust in AI systems, there is debate about what trust of AI *is*, and if unjustified trust can be harmful. A prominent proponent of complete AI trust is the European Commission's High-level Expert Group on AI (HLEG) [28]. This promotion of this trust is seen as harmful by some who believe creating a reliance on AI will have a negative impact on humanity and that AI can never truly be trusted due to its nature [29]. Explainable AI, as presented in this paper, is intended to allow better validation and understanding of models which are in use or proposed for use.

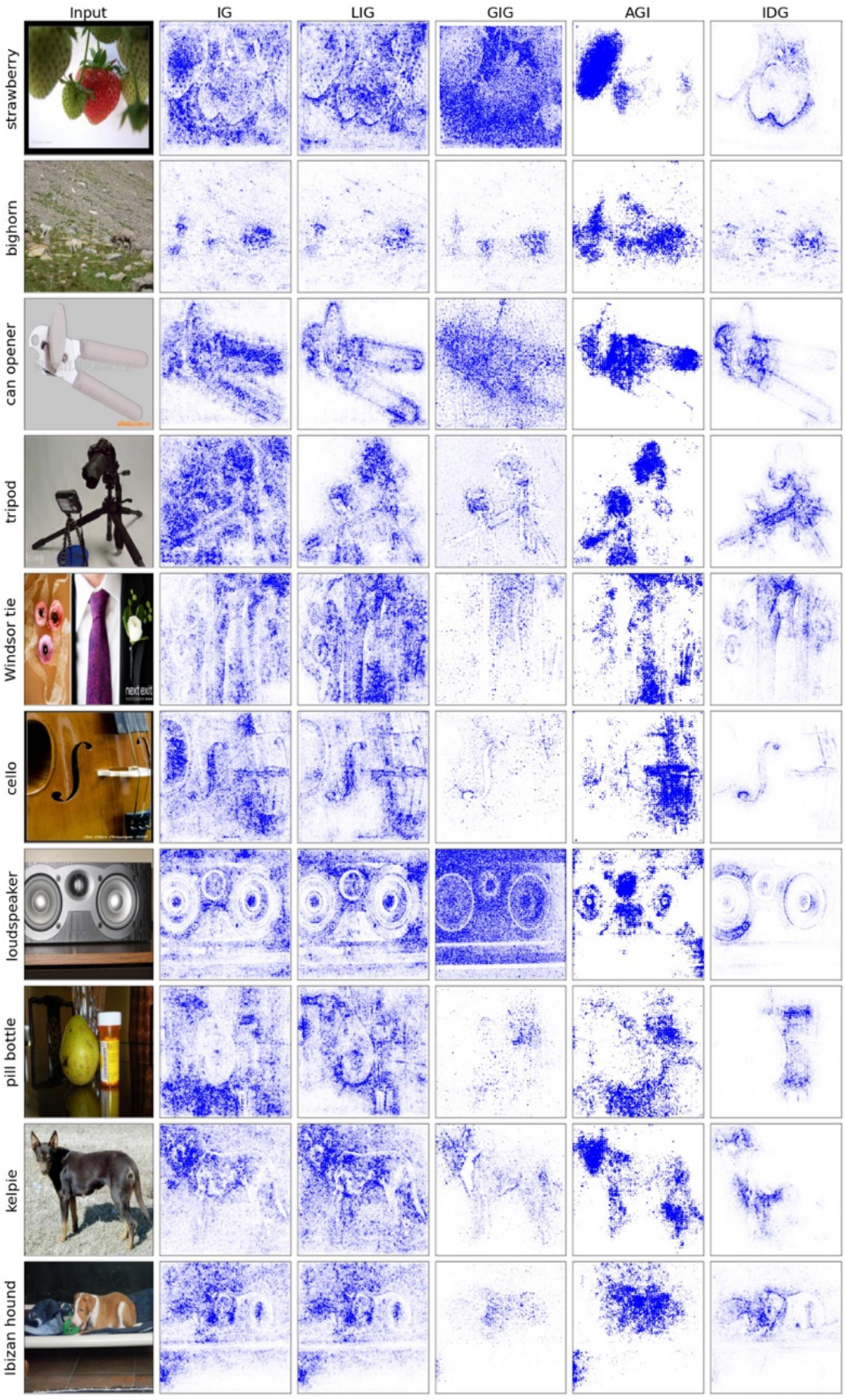
A.5 Additional Visual Comparisons

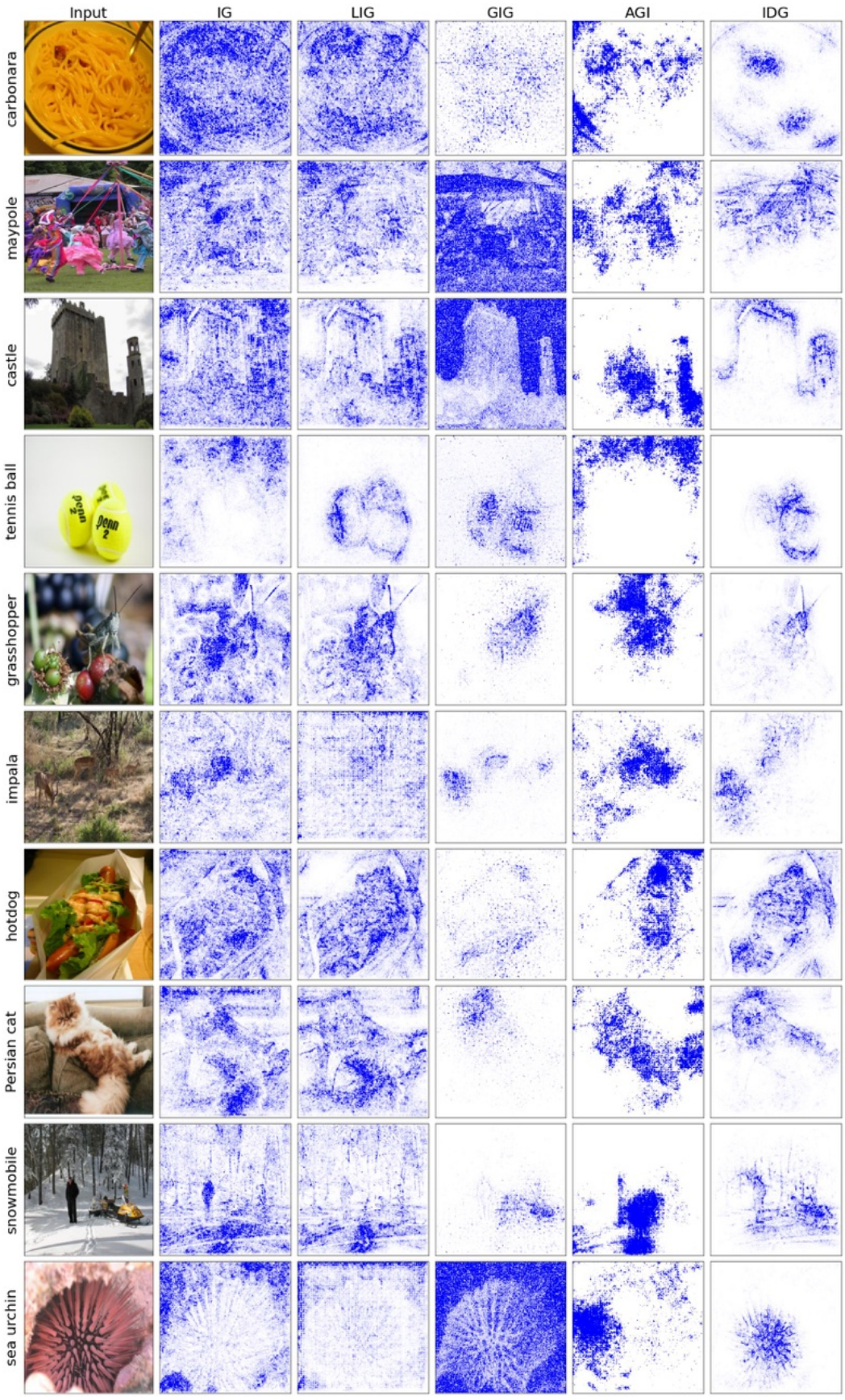
To validate the quantitative performance presented in the paper, we visually compare IG [10], LIG [11], GIG [12], AGI [13], and IDG with a larger number of examples. We present 50 example images on pages 14 - 18. There are six columns per example. From left to right the columns are: the input image, IG, LIG, GIG, AGI, and IDG. These labels are provided above the columns on each page and the class of the input image is provided to its left. The images are from the ImageNet validation set. We use ResNet101 on pages 14 - 16, ResNet152 on page 17, and ResNeXt on page 18 [23, 24]. The attributions are generated with the same parameters as the quantitative testing.

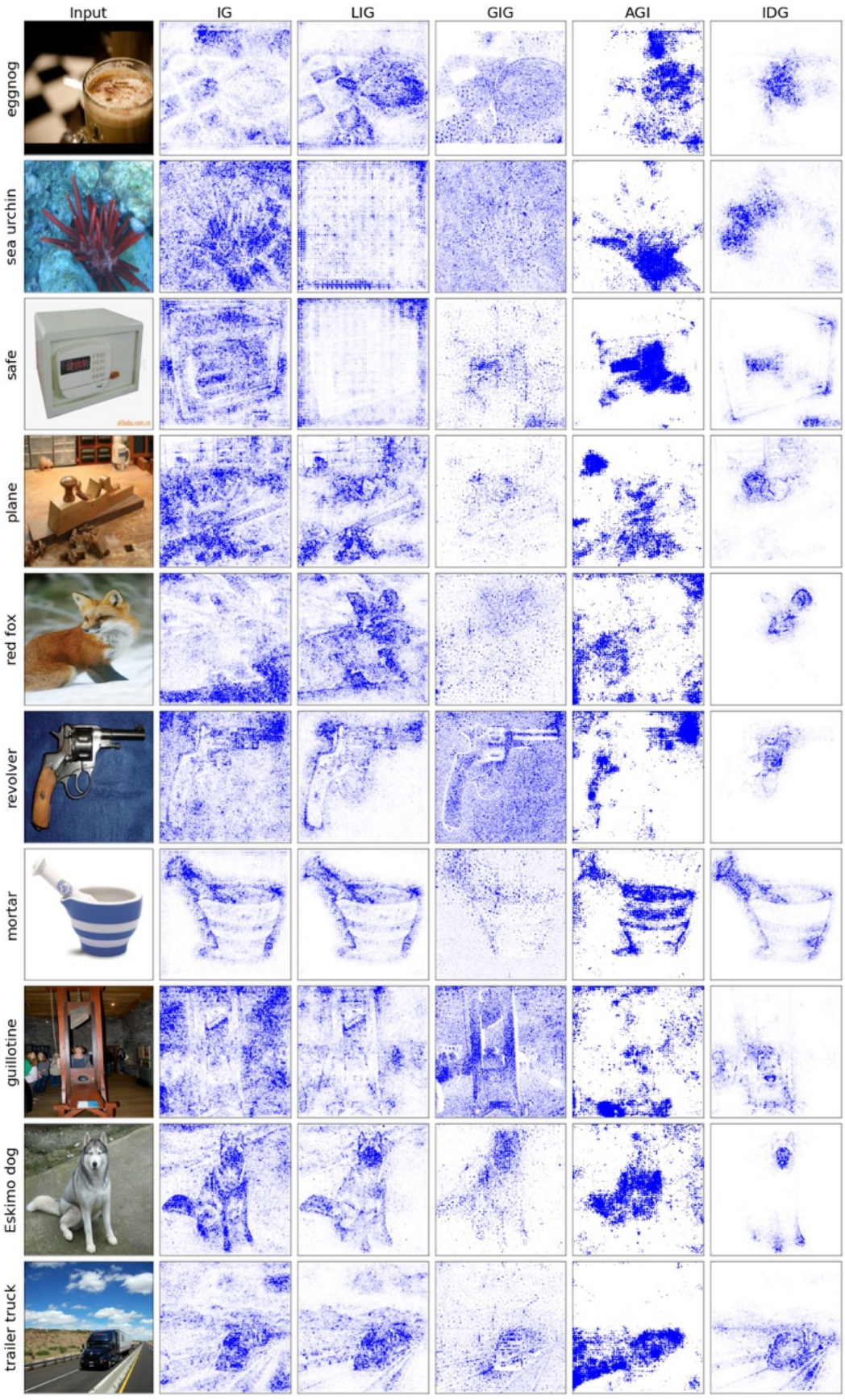
The presented attributions are analyzed visually. A stronger attribution is defined by reduction of noise in areas irrelevant to the object of the image, and stronger attribution (darker color) in areas where the object exists. After visual analysis, we believe IDG presents a sharper attribution than all of the methods presented for a majority of the provided examples. This thorough qualitative analysis provides further proof of the strength of the proposed IDG method.


Input
IG
LIG
GIG
AGI
IDG
soup bowl
crane
Granny Smith
table lamp
half track
abacus
spoonbill
axolotl
switch
cab


Input
IG
LIG
GIG
AGI
IDG
tripod
eel
tusker
kite
black grouse
breastplate
cinema
macaque
wood rabbit
dowitcher


Input
IG
LIG
GIG
AGI
IDG
strawberry
bighorn
can opener
tripod
Windsor tie
cello
loudspeaker
pill bottle
kelpie
Ibizan hound


Input
IG
LIG
GIG
AGI
IDG
carbonara
maypole
castle
tennis ball
grasshopper
impala
hotdog
Persian cat
snowmobile
sea urchin


Input
IG
LIG
GIG
AGI
IDG
eggnog
sea urchin
safe
plane
red fox
revolver
mortar
guillotine
Eskimo dog
trailer truck