# Appendix: Supplementary Materials for Visual Explanations via Iterated Integrated Attributions

## A. Evaluation Metrics

There is no single measure or test set which is generally acceptable for evaluating explanation maps. Hence, in order to ensure comparability, the evaluations in this research follow earlier works [12, 13, 15, 35, 45]. In general, the various tests entail different types of masking of the original input according to the explanation maps and investigating the change in the model's prediction for the masked input compared to its original prediction based on the unmasked input. There are two variants for these tests which differ based on the class of reference. In one variant, the difference in predictions refers to the ground-truth class, and in the second variant, the difference in predictions refers to the model's original top-predicted class. In the manuscript, we report results for both variants and dub the first variant as 'target' and the second variant as 'predicted', respectively.

In what follows, we list and define the different evaluation measures used in this research:

1. Average Drop Percentage **(ADP)** [12]: $\text{ADP} = 100\% \cdot \frac{1}{N} \sum_{i=1}^{N} \frac{\max(0, Y_i^c - O_i^c)}{Y_i^c}$, where $N$ is the total number of images in the evaluated dataset, $Y_i^c$ is the model's output score (confidence) for class $c$ w.r.t. the original image $i$. $O_i^c$ is the same model's score, this time w.r.t. to a masked version of the original image (produced by the Hadamard product of the original image with the explanation map). The **lower** the ADP the better the result.

2. Percentage of Increase in Confidence **(PIC)** [12]: $\text{PIC} = 100\% \cdot \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}(Y_i^c < O_i^c)$. PIC reports the percentage of cases in which the model's output scores increase as a result of the replacement of the original image with the masked version based on the explanation map. The explanation map is expected to mask the background and help the model to focus on the original image. Hence, the **higher** the PIC the better the result.

3. Perturbation tests entail a stepwise process in which pixels in the original image are gradually masked out according to their relevance score obtained from the explanation map [15]. At each step, an additional 10% of the pixels are removed and the original image is gradually blacked out. The performance of the explanation model is assessed by measuring the area under the curve (AUC) with respect to the model's prediction on the masked image compared to its prediction with respect to the original (unmasked) image. We consider two types of masking:

   (a) Positive perturbation (**POS**), in which we mask the pixels in decreasing order, from the highest relevance to the lowest, and expect to see a steep decrease in performance, indicating that the masked pixels are important to the classification score. Hence, for the POS perturbation test, lower values indicate better performance.

   (b) Negative perturbation (**NEG**), in which we mask the pixels in increasing order, from lowest to highest. A good explanation would maintain the accuracy of the model while removing pixels that are not related to the class of interest. Hence, for the NEG perturbation test, lower values indicate better performance.

   In both positive and negative perturbations, we measure the area-under-the-curve (AUC), for erasing between 10%-90% of the pixels. As explained above, results are reported with respect to the 'predicted' or the 'target' (ground-truth) class.

4. The deletion and insertion metrics [45] are described as follows:

   (a) The deletion (**DEL**) metric measures a decrease in the probability of the class of interest as more and more important pixels are removed, where the importance of each pixel is obtained from the generated explanation map. A sharp drop and thus a low area under the probability curve (as a function of the fraction of removed pixels) means a good explanation.

   (b) In contrast, the insertion (**INS**) metric measures the increase in probability as more and more pixels are revealed, with higher AUC indicative of a better explanation.

   Note that there are several ways in which pixels can be removed from an image [16]. In this work, we remove pixels by setting their value to zero. Gradual removal or introduction of pixels is performed in steps of 0.1 i.e., remove or introduce 10% of the pixels on each step).

5. The Accuracy Information Curve (**AIC**) and the Softmax Information Curve (**SIC**) [35] metrics are both similar in spirit to the receiver operating characteristics (ROC). These measures are inspired by the Bokeh effect in photography [40], which consists of focusing on objects of interest while keeping the rest of the image blurred. In a similar fashion, we start with a completely blurred image and gradually sharpen the image areas that are deemed important by a given explanation method. Gradually sharpening the image areas increases the information content of the image. We then compare the explanation methods by measuring the approximate image entropy (e.g., compressed image size) and the model's performance (e.g., model accuracy).

   (a) The AIC metric measures the accuracy of a model as a function of the amount of information provided to the explanation method. AIC is defined as the AUC of the accuracy vs. information plot. The information provided to the method is quantified by the fraction of input features that are considered during the explanation process.

   (b) The SIC metric measures the information content of the output of a softmax classifier as a function of the amount of information provided to the explanation method. SIC is defined as the AUC of the entropy vs. information plot. The entropy of the softmax output is a measure of the uncertainty or randomness of the classifier's predictions. The information provided to the method is quantified by the fraction of input features that are considered during the explanation process.

## B. Baselines Description

1. Grad-CAM (**GC**) [46] integrates the activation maps from the last convolutional layer in the CNN by employing global average pooling on the gradients and utilizing them as weights for the feature map channels.

2. Grad-CAM++ (**GC++**) [12] is an advanced variant of Grad-CAM that utilizes a weighted average of the pixel-wise gradients to generate the activation map weights.

3. XGrad-CAM (**XGC**) [25] calculates activation coefficients using two axioms. Although the authors derived coefficients that satisfy these axioms as closely as possible, their derivation is only demonstrated for ReLU-CNNs.

4. Integrated Gradients (**IG**) [54] integrates over the interpolated image gradients.

5. Blur IG (**BIG**) [59] is concerned with the introduction of information using a baseline and opts to use a path that progressively removes Gaussian blur from the attributed image.

6. Guided IG (**GIG**) [36] improves upon Integrated Gradients by introducing the idea of an adaptive path method. By calculating integration along a different path than Integrated Gradients, high gradient areas are avoided which often leads to an overall reduction in irrelevant attributions.

7. LIFT-CAM (**LIFT**) [34] employs the DeepLIFT [47] technique to estimate the activation maps SHAP values [42] and then combine them with the activation maps to produce the explanation map.

8. The FullGrad (**FG**) method [53] provides a complete modeling approach of the gradient by also taking the gradient with respect to the bias term, and not just with respect to the input.

9. LayerCAM (**LC**) [33] utilizes both gradients and activations, but instead of using the Grad-CAM approach and applying pooling on the gradients, it treats the gradients as weights for the activations by assigning each location in the activations with an appropriate gradient location. The explanation map is computed with a location-wise product of the positive gradients (after ReLU) with the activations, and the map is then summed w.r.t. the activation channel, with a ReLU applied to the result.

10. Ablation-CAM (**AC**) [19] is an approach that only uses the channels of the activations. It takes each activation channel, masks it from the final map by zeroing out all locations of this channel in the explanation map produced by all the channels, computes the score on the masked explanation map (the map without the specific channel), and this score is used to assign an importance weight for every channel. At last, a weighted sum of the channels produces the final explanation map.

11. The Transformer attribution (**T-ATTR**) [15] method computes the importance of each input token by analyzing the attention weights assigned to it during self-attention. Specifically, it computes the relevance score of each token as the sum of its attention weights across all layers of the Transformer. The intuition behind this approach is that tokens that receive more attention across different layers are likely more important for the final prediction. To obtain a more interpretable and localized visualization of the importance scores, the authors also propose a variant of the method called Layer-wise Relevance Propagation (LRP), which recursively distributes the relevance scores back to the input tokens based on their contribution to the intermediate representations.

12. Generic Attention Explainability (**GAE**) [14] is a generalization of T-Attr for explaining Bi-Modal transformers.

## C. Sanity Checks for Explanation Maps

In order to further evaluate the soundness and validity of IIA, we conducted both the *parameter randomization* and *data randomization* sanity tests as proposed by [2]. Unless stated otherwise, the experiments utilize the ImageNet ILSVRC 2012 validation set [18] with the VGG-19 [51] model and IIA3.

### C.1. Parameter Randomization Test

The parameter randomization test compares the explanation maps produced by the explanation method based on two setups of the same model architecture: (1) trained - the model is trained on the dataset (e.g., a pretrained VGG-19 model that was trained on ImageNet, and (2) random - the same model architecture, with random weights (e.g., a randomly initialized VGG-19 model). For a method that relies on the actual model to be explained, we anticipate significant differences in the explanation maps produced for the trained model and those produced for the random model. Conversely, if the explanation maps are similar, we conclude that the explanation method is insensitive to the model's parameters, and thus may not be useful for explaining and debugging the model.

Given a trained model, we consider two types of parameter randomization tests: The first test randomly re-initializes all weights of the model in a cascading fashion (layer after layer). The second test independently randomizes one layer at a time, while keeping all other layers fixed. In both cases, we compare the resulting explanations obtained by using the model with random weights to those derived from the original weights of the model.

**Cascading Randomization** The cascading randomization method involves the randomization of a model's weights, starting from the top layer and successively moving down to the bottom layer. This process leads to the destruction of the learned weights from the top to the bottom layers. Figure 5 presents the Spearman correlation (averaged on 50K examples) between the original explanation map obtained by IIA and the original (pretrained) VGG-19 model and the explanation map obtained by IIA and each of the cascade randomization versions of the original model. The markers on the x-axis are between '0' and '16', where $x = k$ means that the weights of the last $k$ layers of the model are randomized. At $x = 0$ there is no randomization, hence the correlation with the original model is perfect. Starting from $x = 1$ (marked by the horizontal dashed line) and up to $x = 16$, the graph depicts a progressive cascade randomization of the original model. We observe that as more layers' weights are randomized, the correlation with the explanation map of the original model significantly deteriorates. This behavior showcases the sensitivity of IIA to the model's parameters - an expected and desired property for any explanation method [2].

Figure 6 displays a representative example of explanation maps (bottom) and their overlay to the original image (top), illustrating the cascading randomization process. The first column presents explanation maps produced by IIA and the original model, while the rest of the columns present explanation maps produced by IIA and cascading randomized models, where the number $i$ above each column indicates that the explanation map is produced by a model in which the weights of the last $i$ layers were randomized. It is evident that the quality of produced explanation maps significantly degrades as more and more layers are set with random weights.

**Independent Randomization** We further consider another version of the model's parameters randomization test, in which a layer-by-layer randomization is employed, one layer at a time. In this test, we aim to isolate the influence of the randomization of each layer, hence randomization is applied to one layer's weights at a time, while all other layers' weights are kept identical to their values in the original model. This randomization methodology enables comprehensive evaluation of the sensitivity of the explanation maps w.r.t. each of the model's layers.

Figure 7 presents results for the independent randomization tests. At $x = 0$ no randomization was applied and the correlation to the original model is perfect. For $x = i$ ($i > 0$) the graph indicates the correlation of the original model with a model in which only the weights of the $i$-th penultimate layer were randomized while the weights of all other layers were

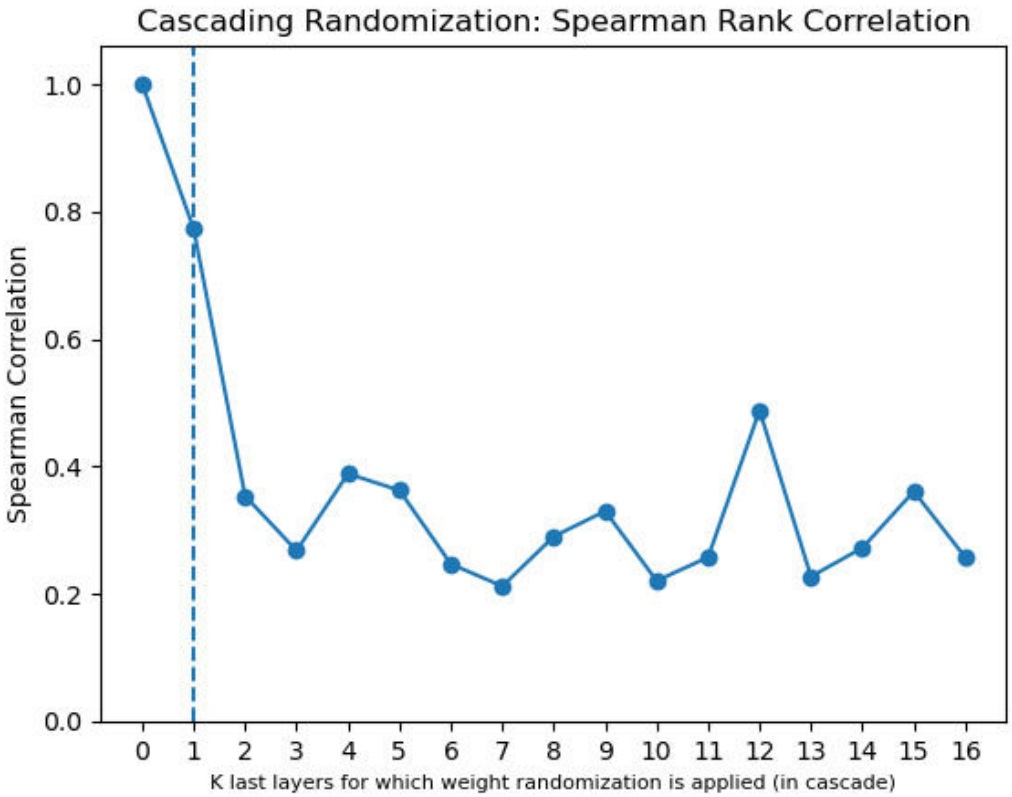


Figure 5. **Cascading Randomization**: The VGG-19 model is subjected to successive weights randomization, beginning from the last model's layers on the ImageNet dataset. The presented graph depicts the Spearman rank correlation (averaged on 50K examples) between the explanation produced by IIA using the original and randomized model's weights. The x-axis corresponds to the number of layers being randomized, starting from the output layer. The dashed line indicates the point where the successive randomization of the network commences, which is at the top layer. The first dot (x=0) corresponds to no randomization (the original model is used), hence the correlation between the explanation maps is perfect. See Sec. C.1 for further details.

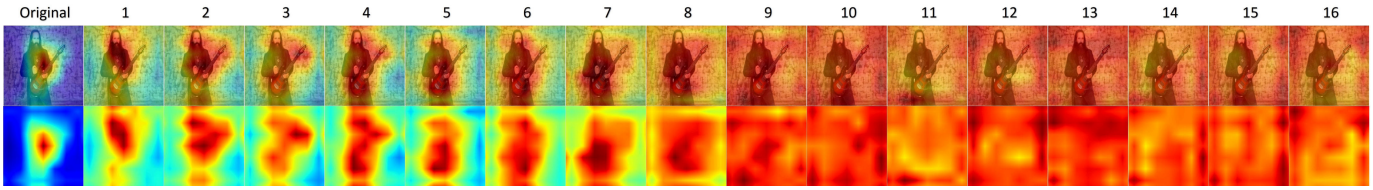


Figure 6. **Cascading Randomization on VGG-19 (ImageNet)**: The figure presents the original explanations (first column) for 'electric guitar'. Progression from left to right depicts the gradual randomization of network weights up to the layer number depicted at the top of the column (starting from the last layer). See Sec. C.1 for further details.

kept untouched. We observe that the correlation values are low across all layers which indicates IIA's sensitivity to weight randomization in each layer separately. This property is a desired property for an explanation method, as it indicates the method's sensitivity to each of the model's layers, independently. Finally, Fig. 8 presents a qualitative example in the same fashion as Fig. 6, this time for the independent randomization test. We observe that the quality of all explanation maps produced by a randomized version of the model differs significantly from the original explanation map. We conclude that IIA successfully passes both types of parameter randomization tests.

## C.2. Data Randomization Test

The data randomization sanity test is a method used to assess whether an explanation method is sensitive to the labeling of the data used for training the model. This is done by comparing the explanation maps produced by the explanation method for

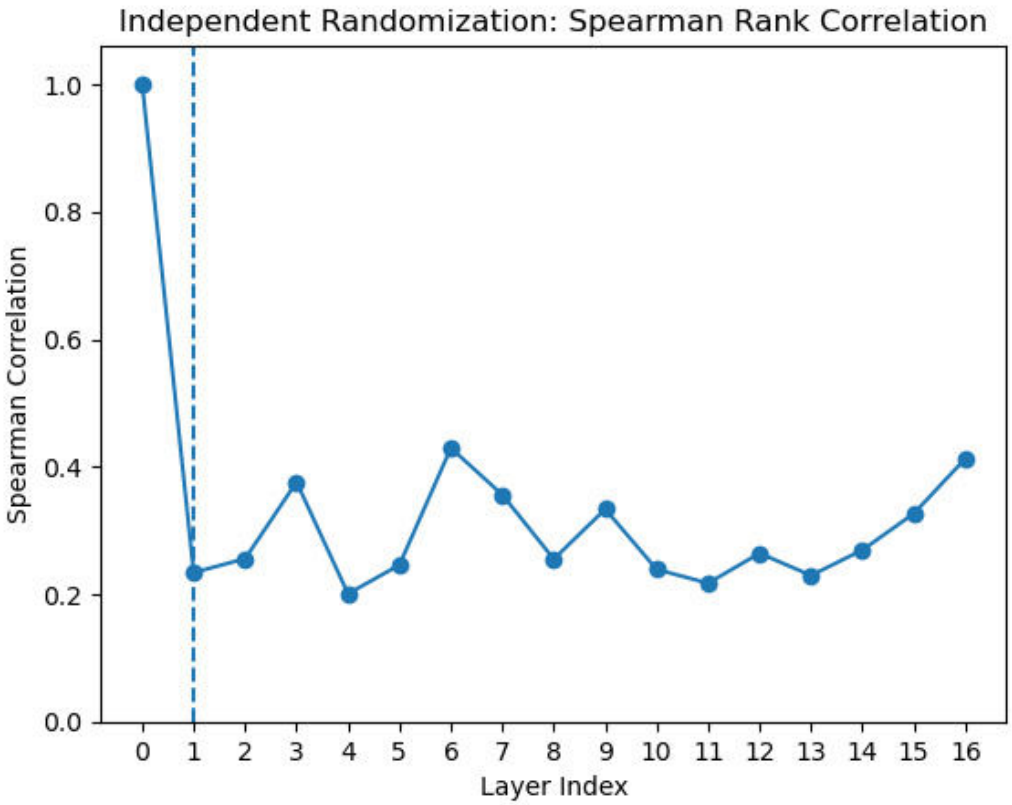


Figure 7. **Independent Randomization**: The randomization process is carried out independently for each layer of the model, while the remaining weights are retained at their pretrained values. The y-axis of the presented graph represents the rank correlation between the original and randomized explanations, with each point on the x-axis corresponding to a specific layer of the model. The dashed line marks the point where the randomization of the network layers commences, which is at the top layer.

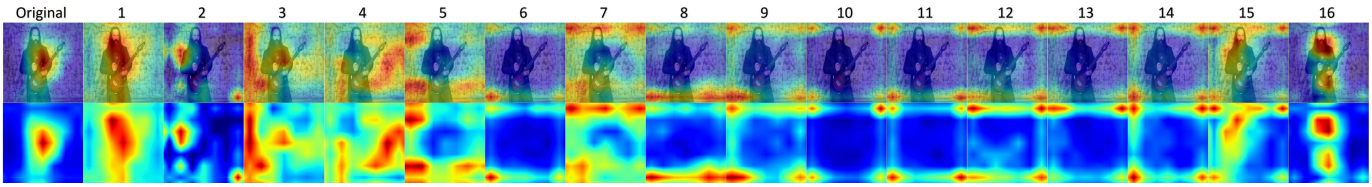


Figure 8. **Independent Randomization on VGG-19 (ImageNet)**: Similar to Fig. 6, however, this time, each specific layer is randomized independently, while the rest of the weights are kept at their pretrained values.

two models with identical architecture that were trained on two different datasets: one with the original labels and another with randomly permuted labels. If the explanation method is sensitive to the labeling of the dataset, we would expect the produced explanation maps to differ significantly between the two cases. However, if the method is insensitive to the permuted labels, it indicates that it does not depend on the relationship between instances and labels that exists in the original data. To conduct the data randomization test, we permute the training labels in the dataset and train the model to achieve a training set accuracy greater than 95%. Note that the resulting model's test accuracy is never better than randomly guessing a label. We then compute explanations on the same test inputs for both the model trained on true labels and the model trained on randomly permuted labels. Figure 9 presents a box plot computed for the Spearman correlation values obtained for paired explanation maps (50K examples): one produced using the original model that is trained with the ground truth, and another produced by the model trained with the permuted labels. We can see that the correlation values are very low indicating IIA's sensitivity to the labeling of the dataset. Hence, we conclude that IIA successfully passes the data randomization test.

Finally, Figure 10 presents additional qualitative examples for both tests, this time with different models. The first row shows two explanation maps produced by IIA w.r.t. the "tabby cat" class. We see that when IIA utilizes an ImageNet pretrained

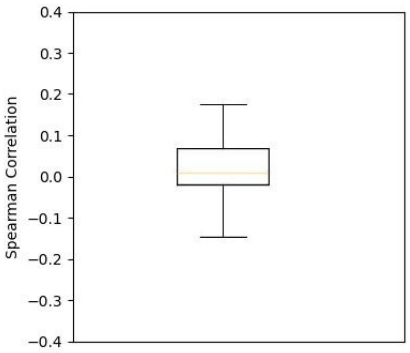


Figure 9. **Data Randomization Test**: Spearman rank correlation box plot for IIA with the VGG-19 model.

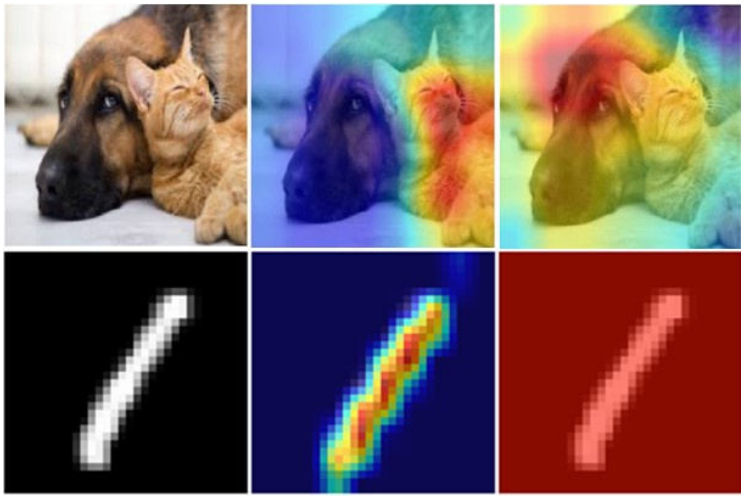

Figure 10. Sanity checks. Rows 1 and 2 present IIA results for the *parameter randomization* and *data randomization* tests w.r.t. the "tabby cat" (ImageNet) and "one" (MNIST) classes, using ResNet50 and LeNet-5, respectively. Left to right: Row 1: Original image, explanation map produced by IIA and the trained model, explanation map produced by IIA and untrained model (model's weights are randomly initialized without further training). Row 2: Original image, explanation map produced by IIA and a model trained with the ground truth labels, explanation map produced by IIA and a model trained with random labels.

ResNet50 model, it produces a focused explanation map (around the cat), but when applying IIA to the same model with random weights, it fails to detect the cat in the image. The second row shows that IIA produces an adequate explanation map when the model (LeNet-5 [38]) is trained with the MNIST ground truth labels but fails when the model is trained with random labels.

## D. Gradient Rollout Implementation

The Gradient Rollout (**GR**) technique is a modified version of the Attention Rollout (**AR**) [1] method, which differentiates itself by including a Hadamard product between each attention map and its gradients in the computation, rather than relying solely on the attention map. The GR method can be expressed mathematically as follows:

$$A'_b = I + E_h(A_b \circ G_b), \tag{11}$$

$$GR = A'_1 \cdot A'_2 \cdots A'_B. \tag{12}$$

where $A_b$ is a 3D tensor consisting of the 2D attention maps produced by each attention head in the transformer block $b$, $G_b$ is the gradients w.r.t. $A_b$. $I$ is the identity matrix, $B$ is the number of transformer blocks in the model, $E_h$ is the mean reduction operation (taken across the attention heads dimension), and $\circ$ and $\cdot$ are the Hadamard product and matrix multiplication operators, respectively.

## E. Additional Qualitative Results

Figures 11-17 present qualitative comparisons between our IIA method (IIA3), T-Attr [15], and GAE [14] (using the ViT-B model). Figures 18-24 present qualitative comparisons between our IIA method (IIA3) and the best-performing methods from Tab. 1 (using the ConvNext model). The explanation maps are produced based on a random set of images sampled for various classes from the IN dataset. Arguably, IIA produces the most accurate explanation maps w.r.t. to the target classes both for CNNs and ViTs.

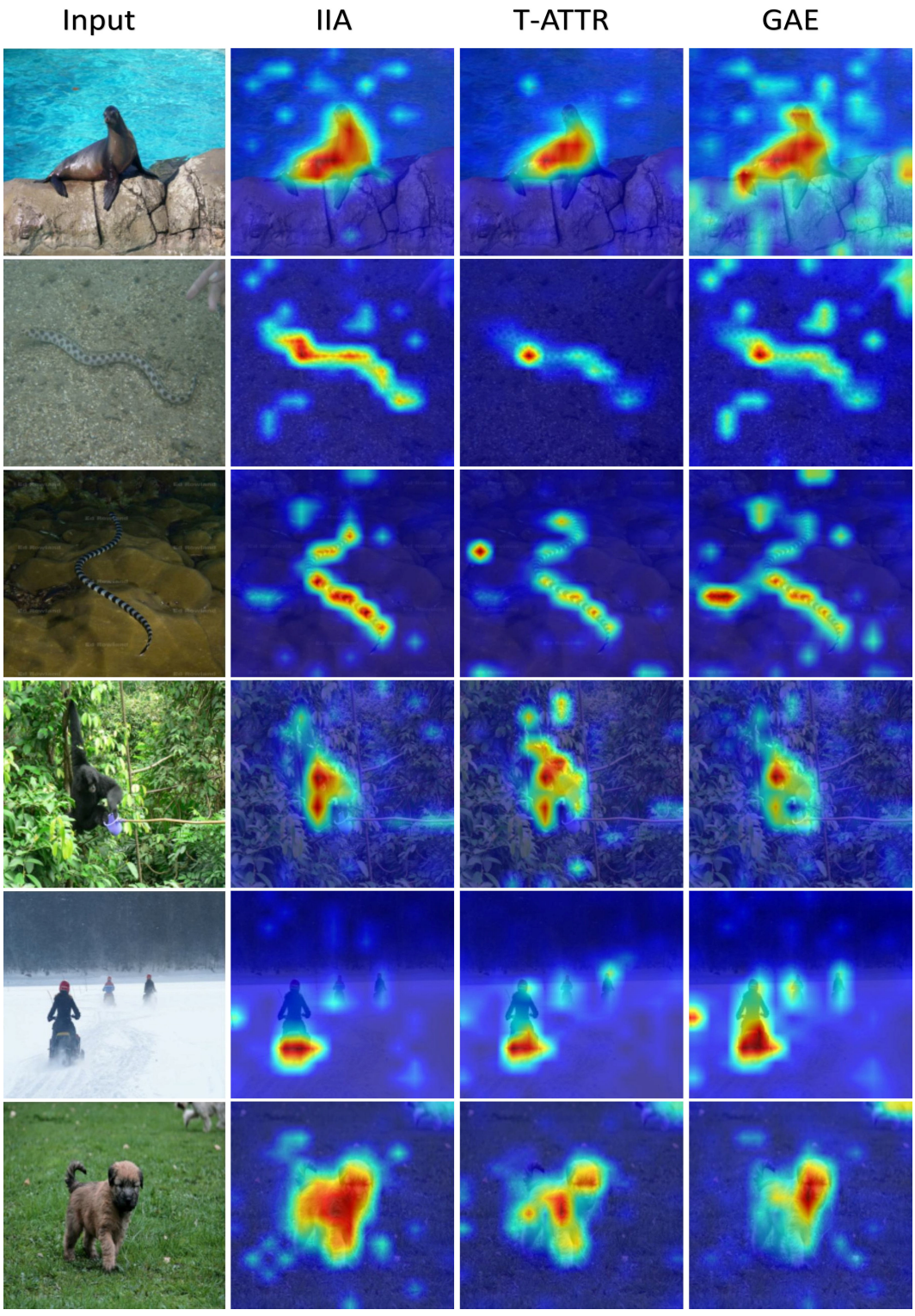


Figure 11. Visualizations obtained by explanation methods for ViT-B model. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1) 'sea lion', (2-3) 'sea snake', (4) 'siamang, Hylobates syndactylus, Symphalangus syndactylus', (5) 'snowmobile', (6) 'soft-coated wheaten terrier'.

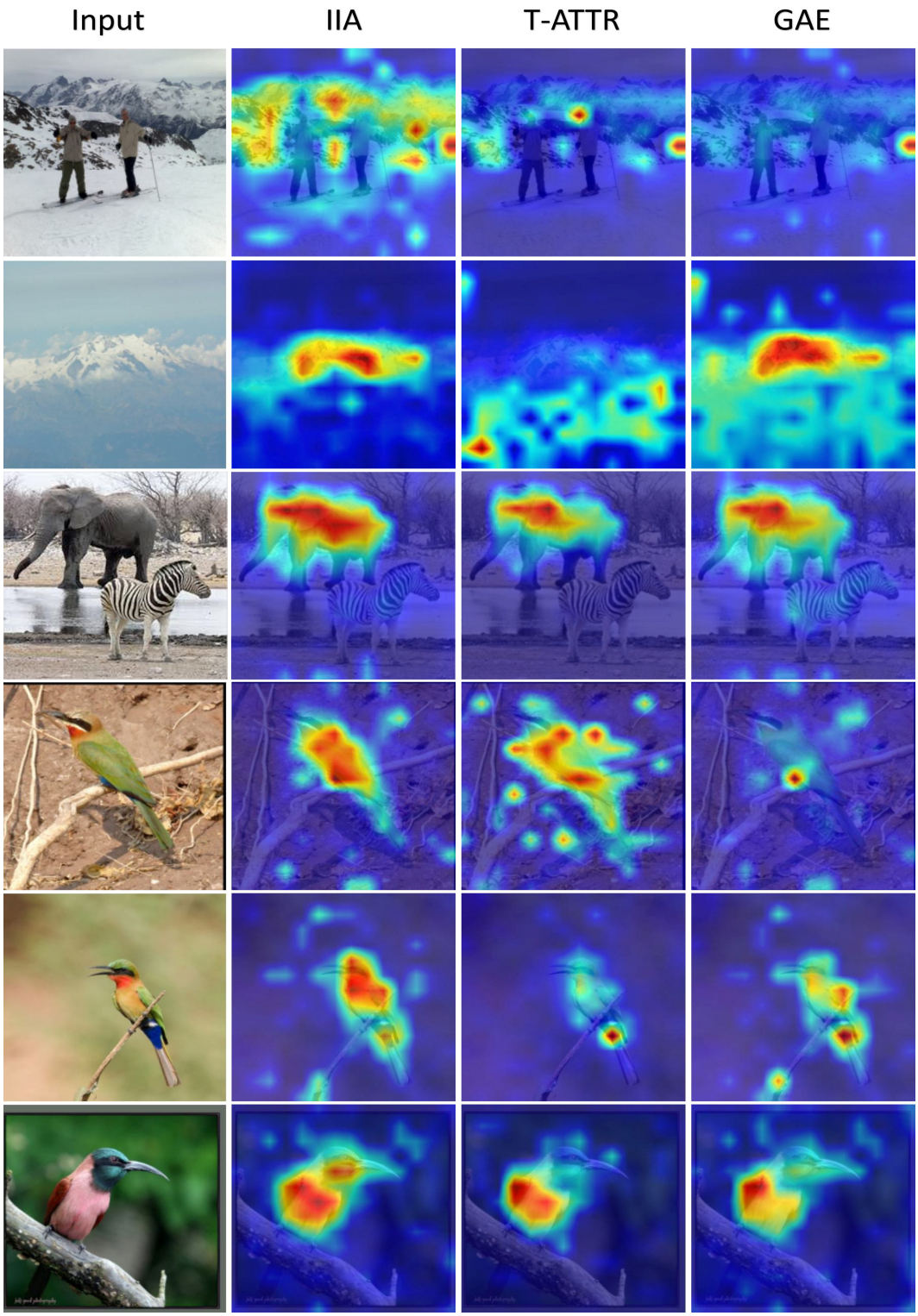


Figure 12. Visualizations obtained by explanation methods for ViT-B model. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1-2) 'alp', (3)'Indian elephant, Elephas maximus', (4-6) 'bee eater'.

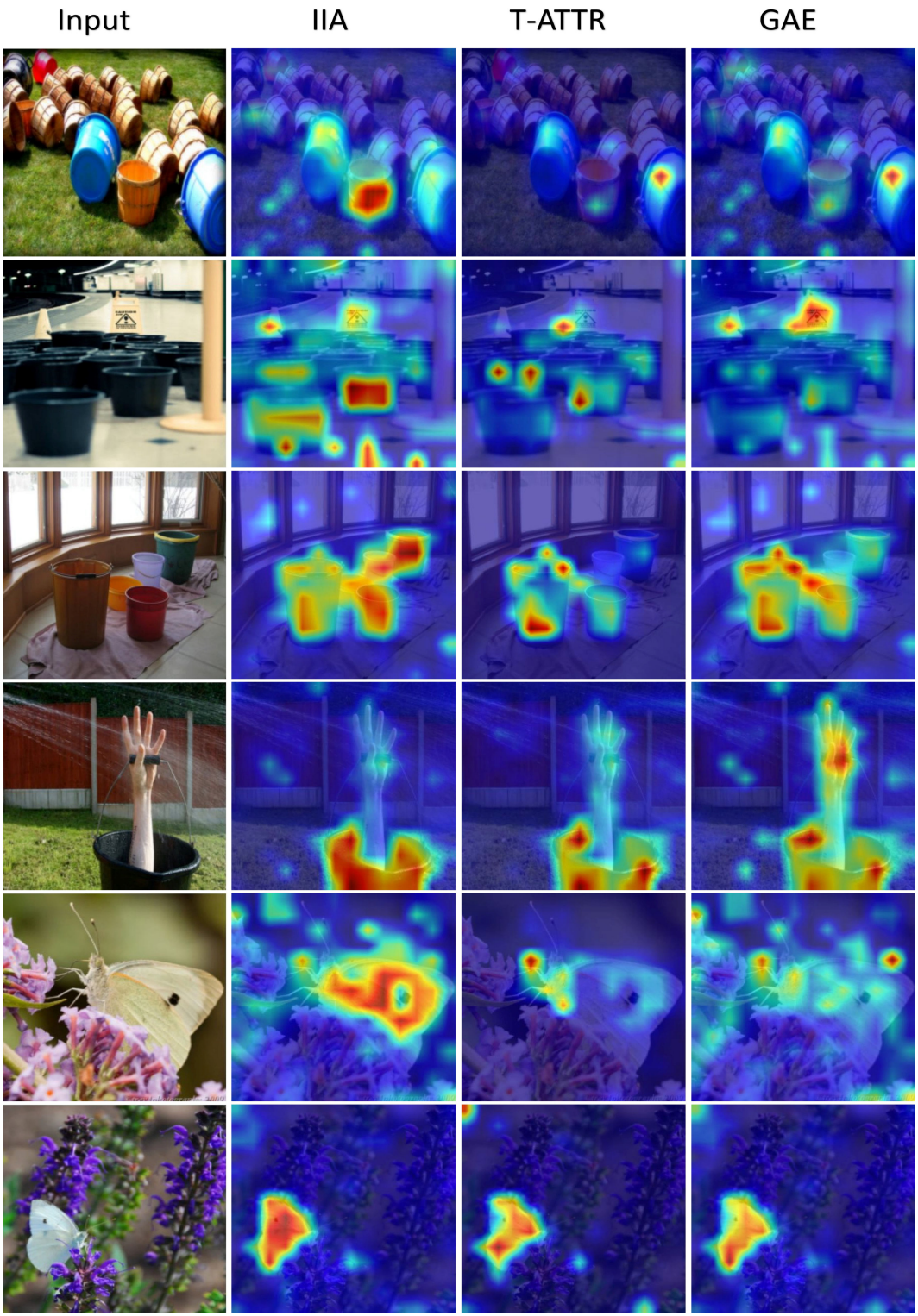


Figure 13. Visualizations obtained by explanation methods for ViT-B model. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1-4) 'bucket, pail', (5-6) 'cabbage butterfly'.

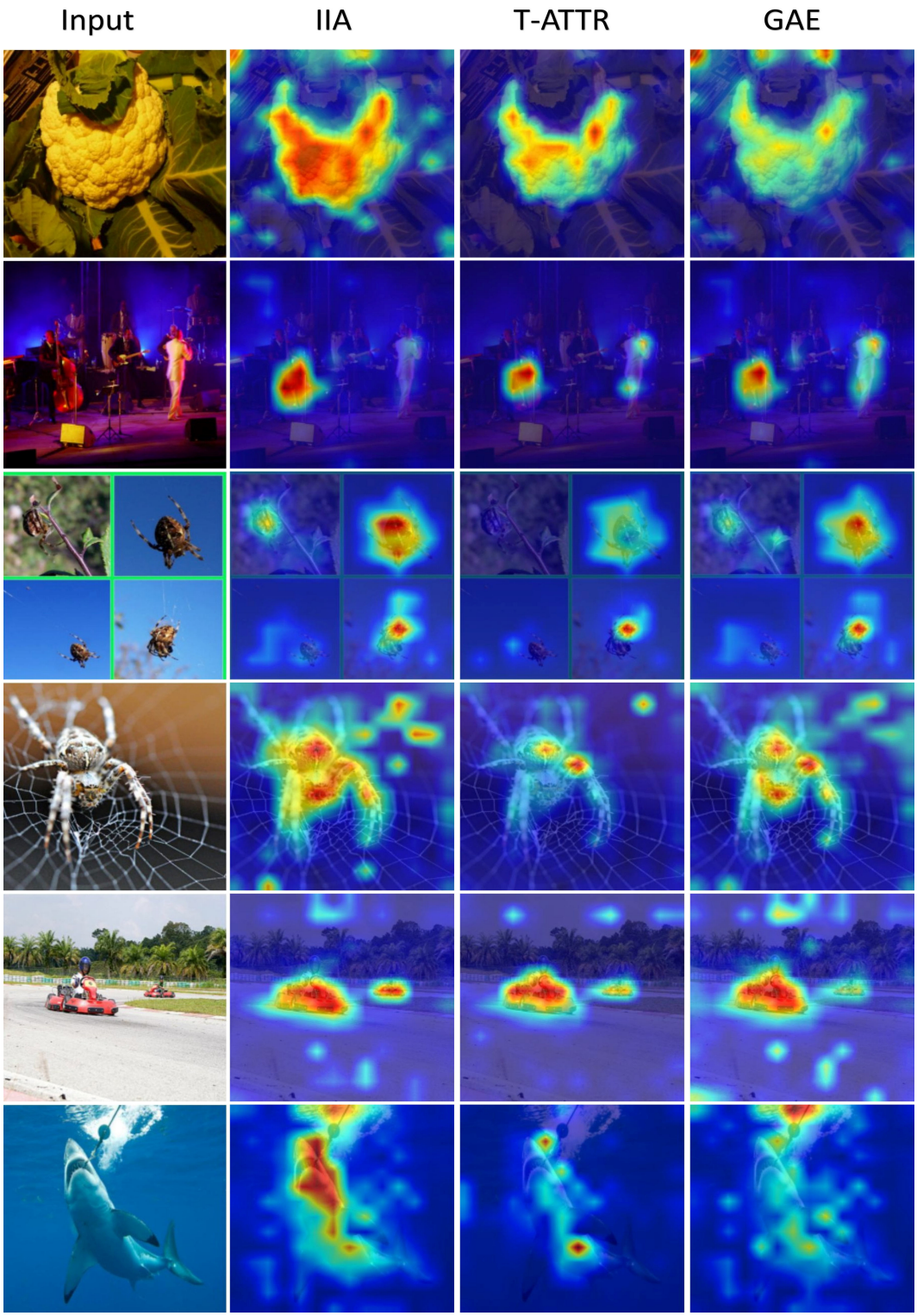


Figure 14. Visualizations obtained by explanation methods for ViT-B model. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1) 'cauliflower', (2) 'cello, violoncello', (3-4) 'garden spider, Aranea diadematа', (5) 'go-kart', (6) 'great white shark, white shark, man-eater, man-eating shark, Carcharodon carcharias'.

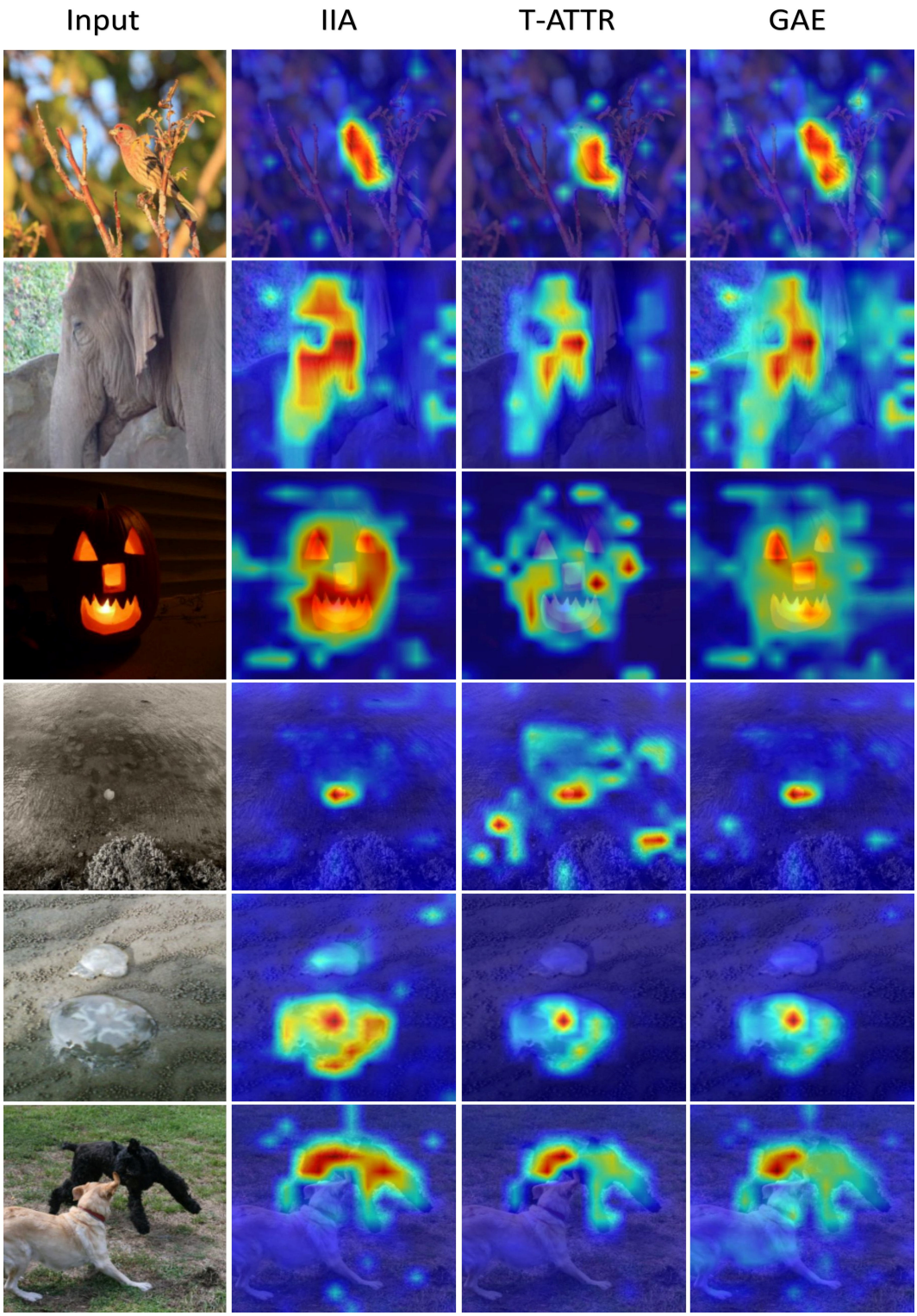


Figure 15. Visualizations obtained by explanation methods for ViT-B model. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1) 'house finch, linnet, Carpodacus mexicanus', (2) 'Indian elephant, Elephas maximus', (3) 'jack-o'-lantern', (4-5) 'jellyfish', (6) 'Kerry blue terrier'.

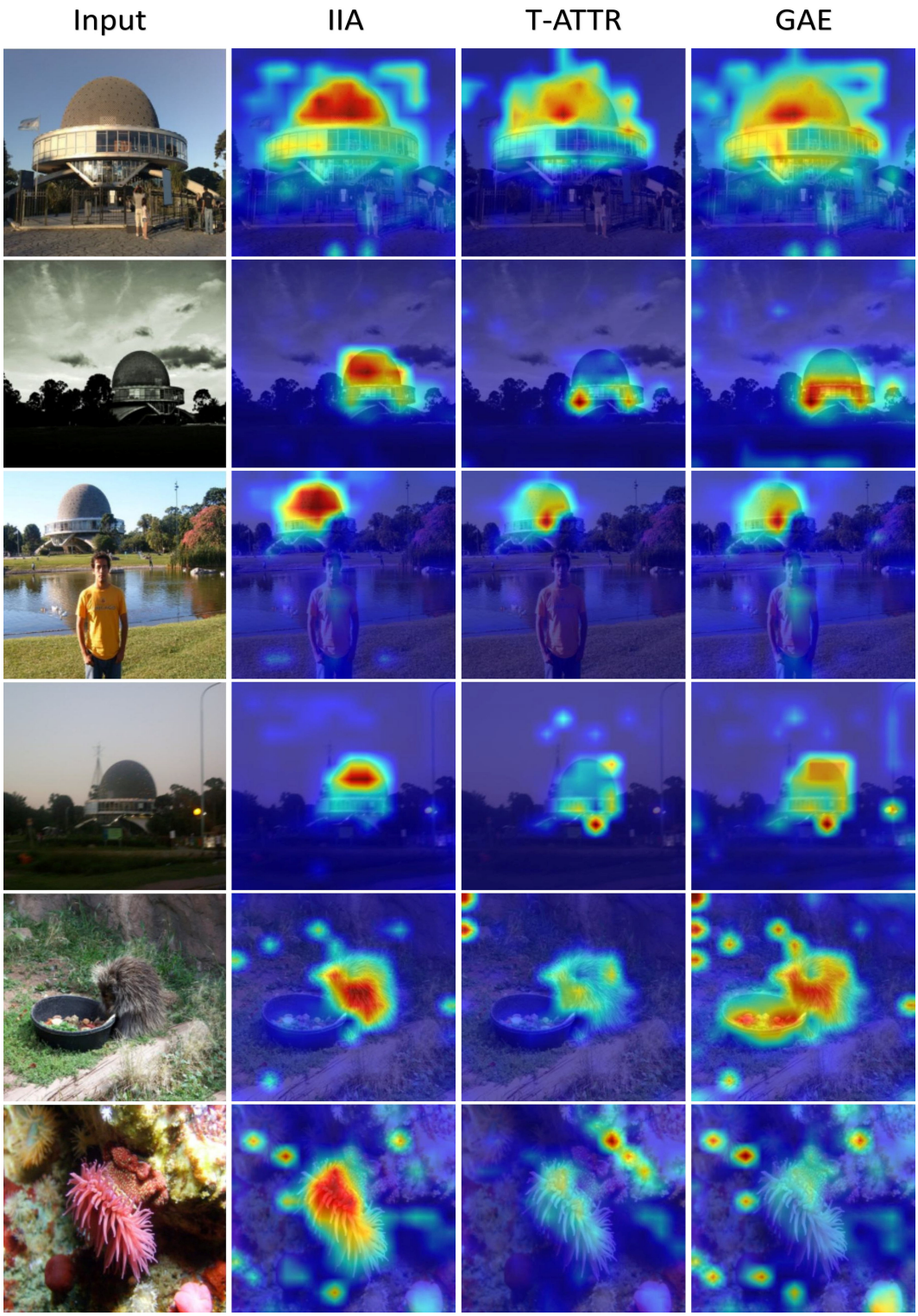


Figure 16. Visualizations obtained by explanation methods for ViT-B model. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1-4) 'planetarium', (5) 'porcupine, hedgehog', (6) 'sea anemone, anemone'.

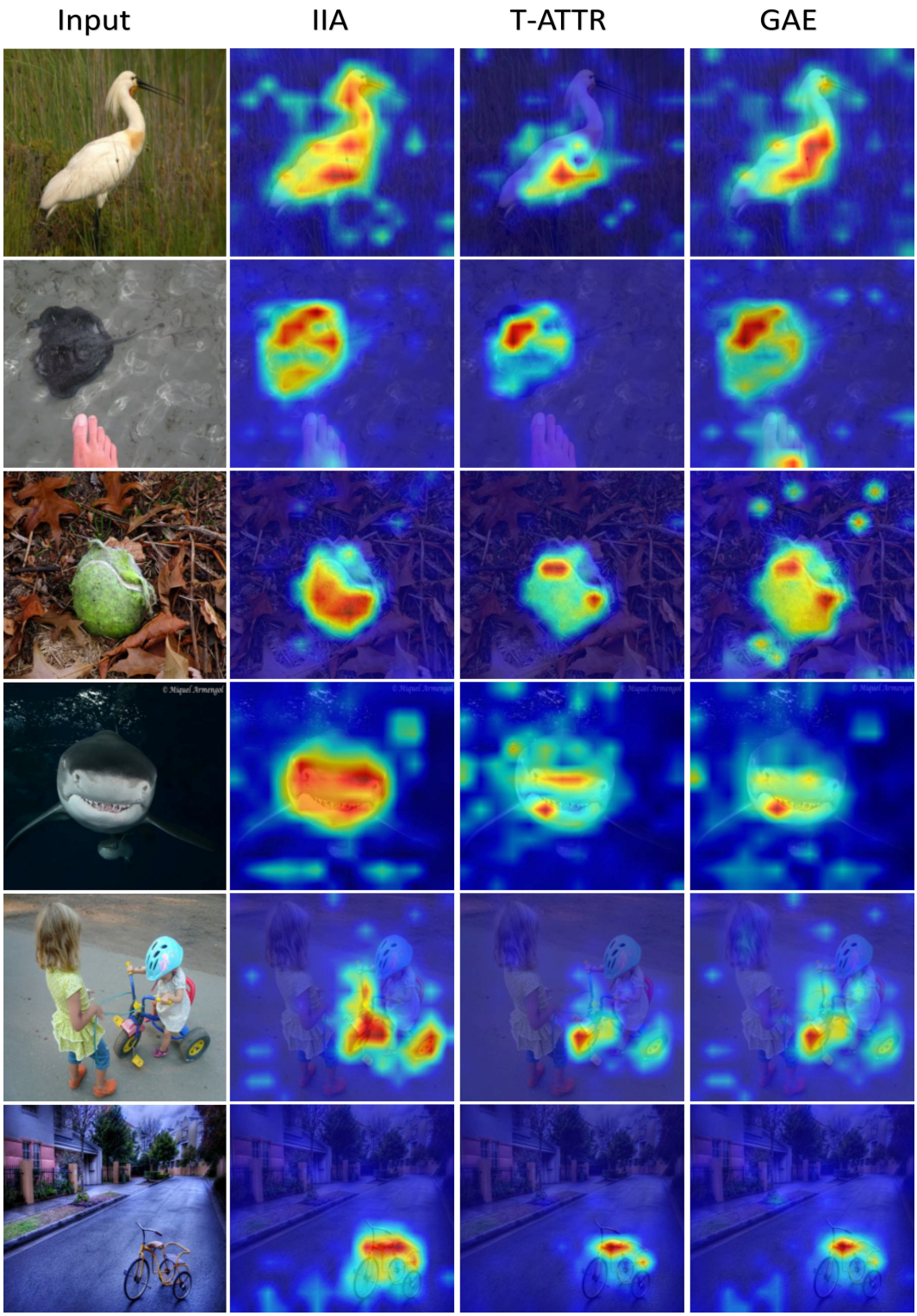


Figure 17. Visualizations obtained by explanation methods for ViT-B model. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1) 'spoonbill', (2) 'stingray', (3) 'tennis ball', (4) 'tiger shark, Galeocerdo cuvieri', (5-6) 'tricycle, trike, velocipede'.

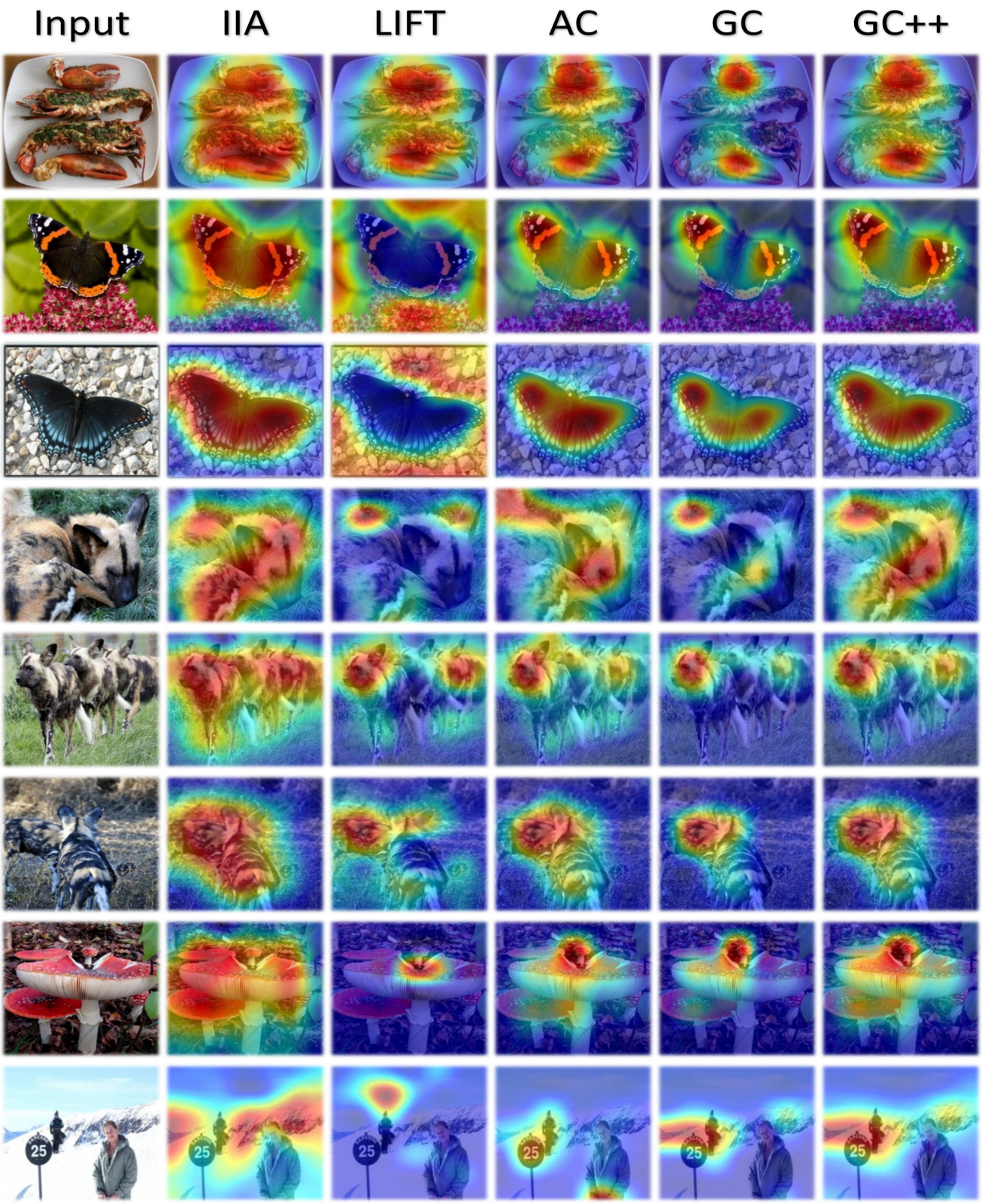


Figure 18. Visualizations obtained by the top performing methods in our evaluations. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1) 'American lobster, Northern lobster, Maine lobster, Homarus americanus', (2,3) 'admiral', (4-6) 'African hunting dog, hyena dog, Cape hunting dog, Lycaon pictus', (7) 'agaric', (8) 'alp'.

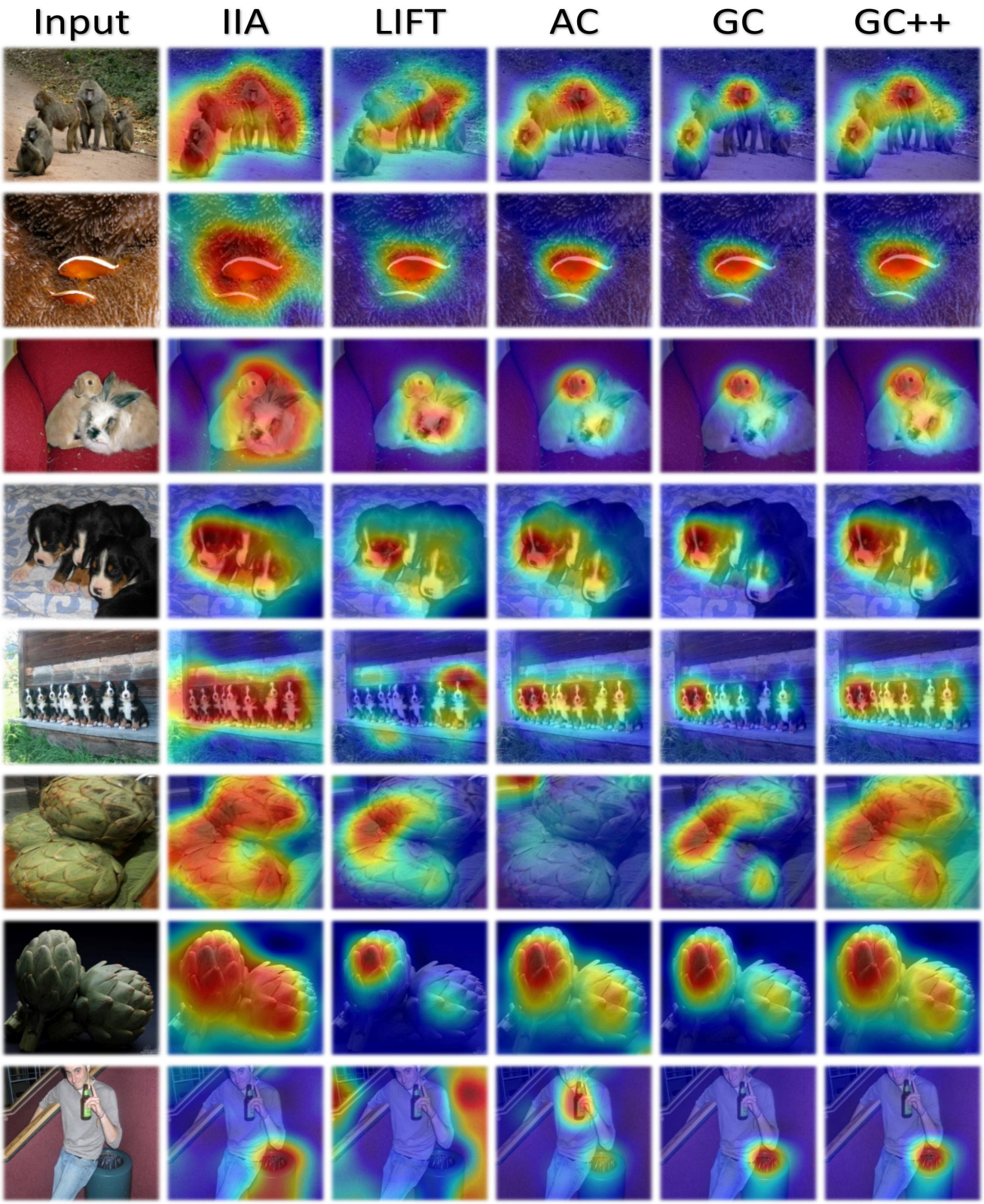


Figure 19. Visualizations obtained by the top performing methods in our evaluations. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1) 'baboon', (2) 'anemone fish', (3) 'Angora, Angora rabbit', (4,5) 'Appenzeller', (6,7) 'artichoke, globe artichoke', (8) 'ashcan, trash can, garbage can, wastebin, ash bin, ash-bin, ashbin, dustbin, trash barrel, trash bin'.

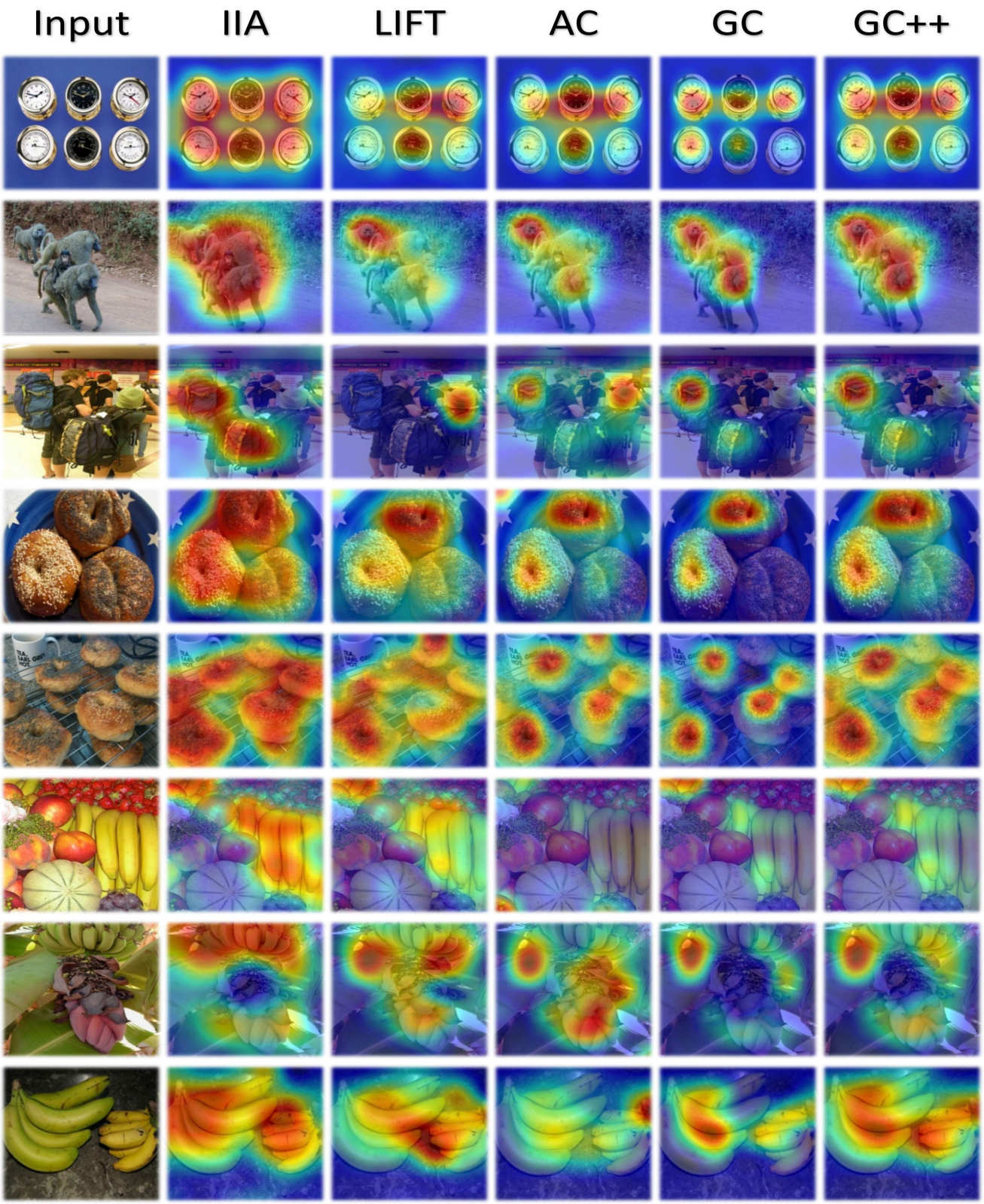


Figure 20. Visualizations obtained by the top performing methods in our evaluations. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1) 'barometer', (2) 'baboon', (3) 'backpack, back pack, knapsack, packsack, rucksack, haversack', (4,5) 'bagel, beigel', (6-8) 'banana'.

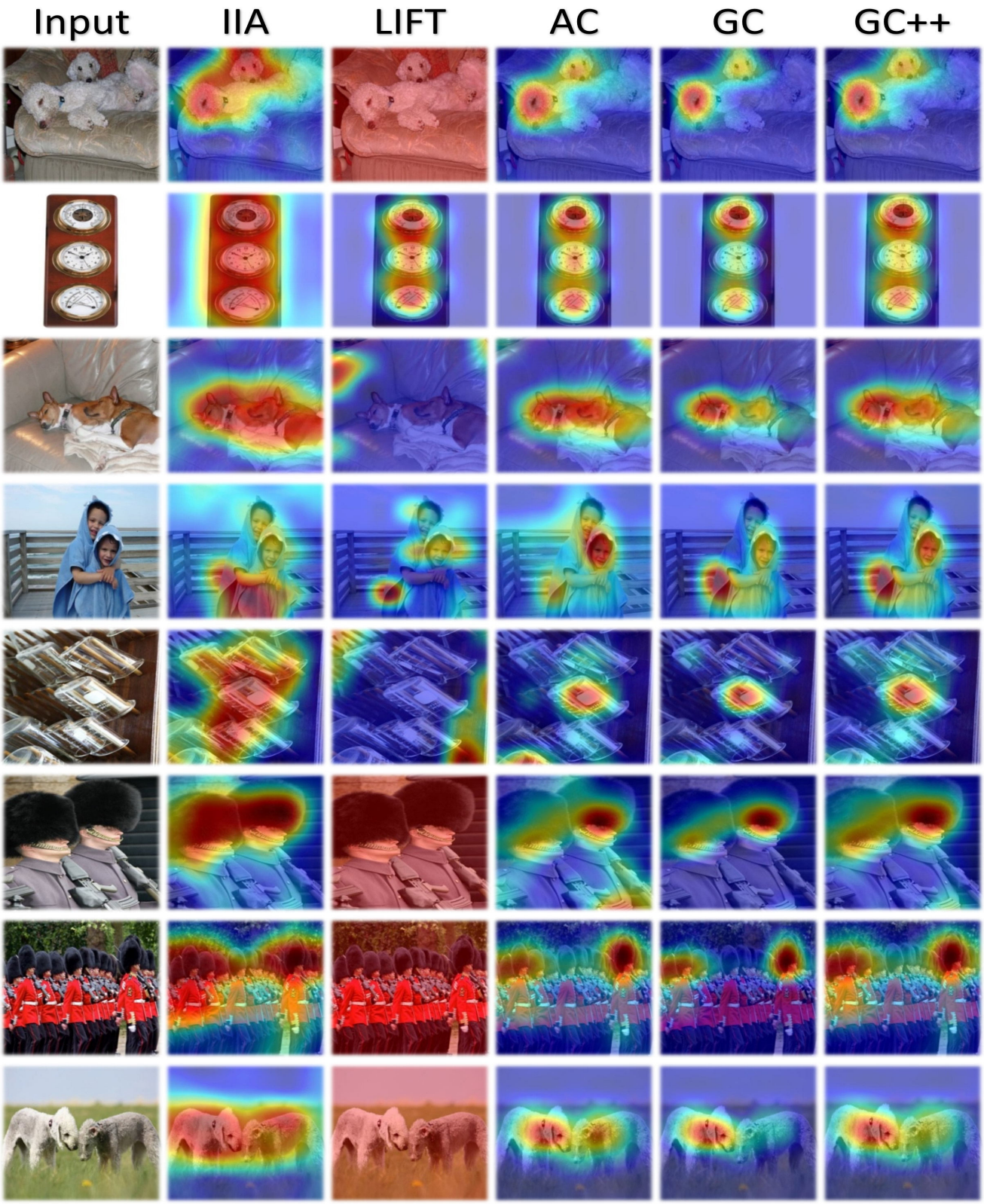


Figure 21. Visualizations obtained by the top performing methods in our evaluations. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1,8) 'Bedlington terrier', (2) 'barometer', (3) 'basenji', (4) 'bath towel', (5) 'beaker', (6,7): 'bearskin, busby, shako'.

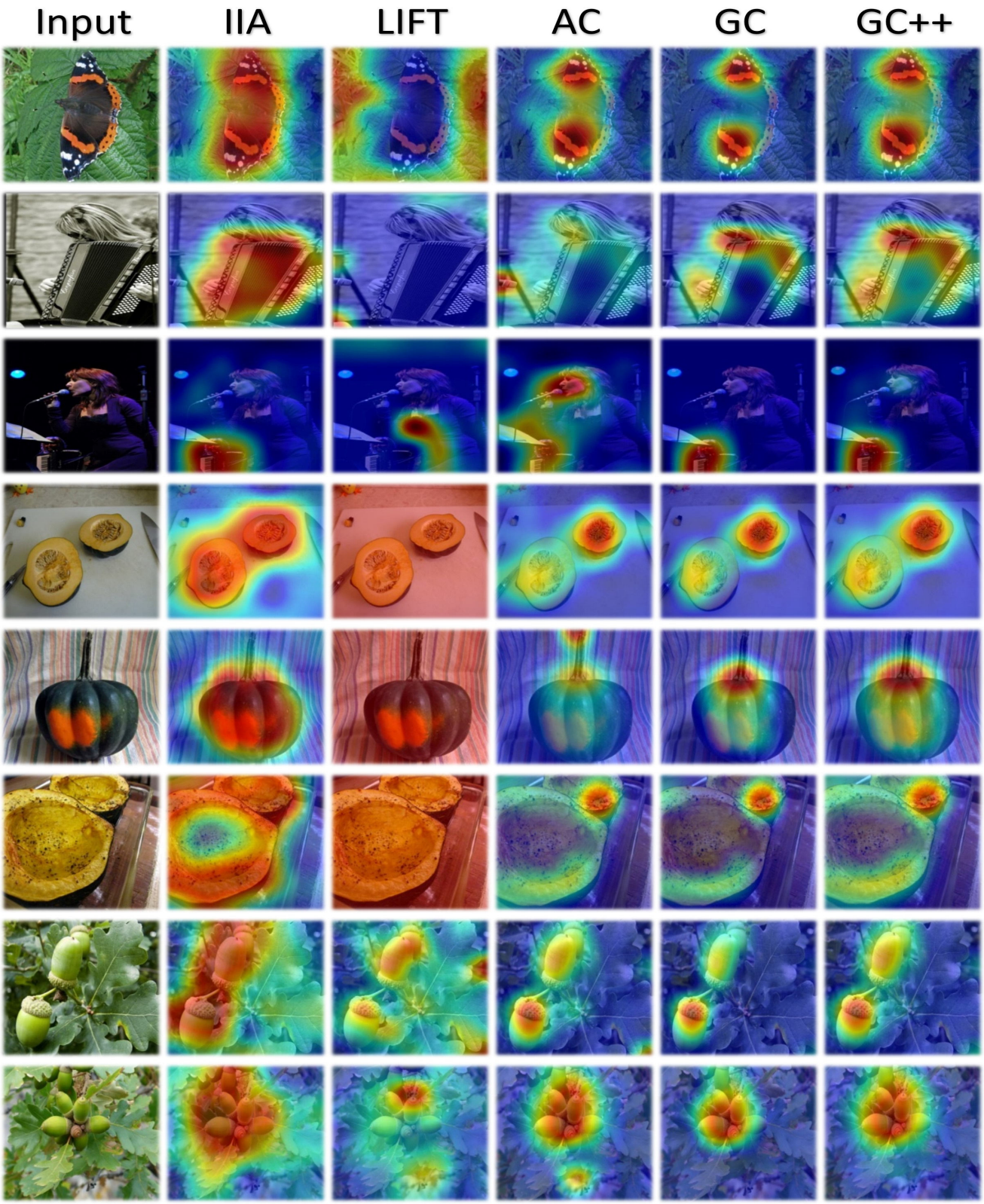


Figure 22. Visualizations obtained by the top performing methods in our evaluations. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1) 'admiral', (2,3) 'accordion, piano accordion, squeeze box', (4-6) 'acron squash', (7,8) 'acron'.

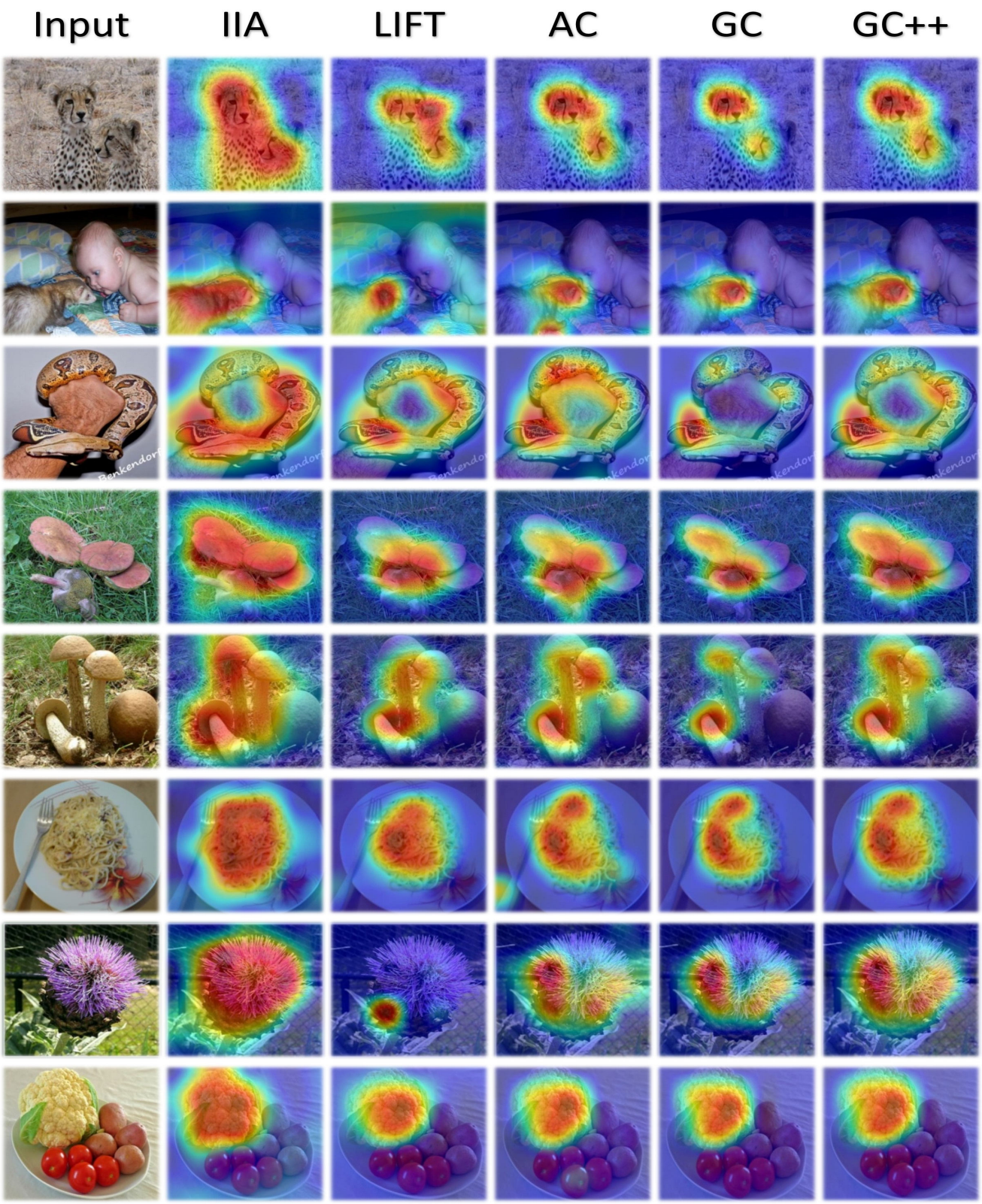


Figure 23. Visualizations obtained by the top performing methods in our evaluations. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1) 'cheetah, chetah, Acinonyx jubatus', (2) 'black-footed ferret, ferret, Mustela nigripes', (3) 'boa constrictor, Constrictor constrictor', (4,5) 'bolete', (6) 'carbonara', (7) 'cardoon', (8) 'cauliflower'.

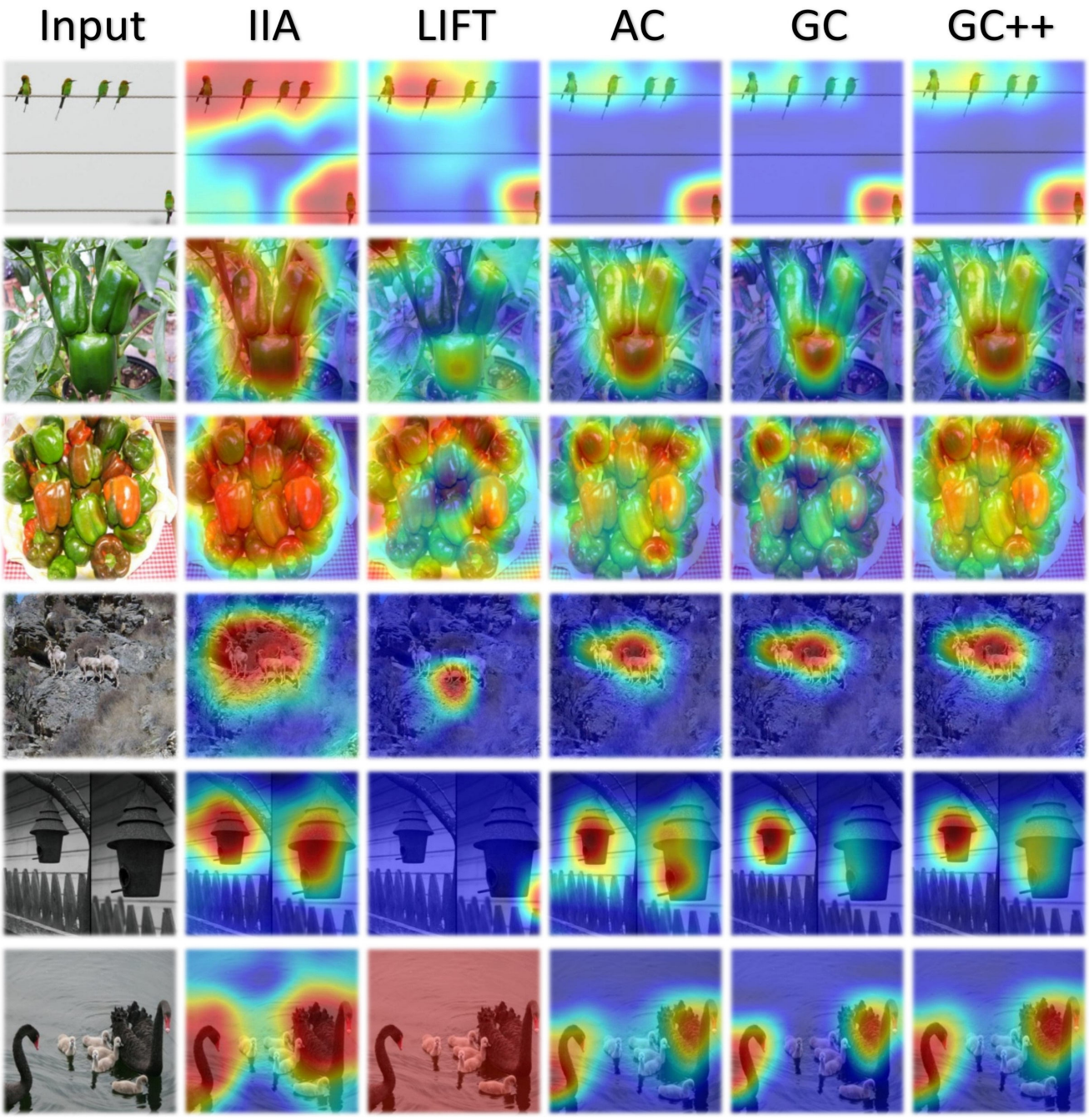


Figure 24. Visualizations obtained by the top performing methods in our evaluations. The ground-truth labels of the images are listed according to the format '(⟨row#⟩) ⟨class names⟩': (1) 'bee eater', (2,3) 'bell pepper', (4) 'bighorn, bighorn sheep, cimarron, Rocky Mountain bighorn, Rocky Mountain sheep, Ovis canadensis', (5) 'birdhouse', (6) 'black swan, Cygnus atratus'.