Appendix: Supplementary Materials for Explainer-Explained Architecture for Vision Models

1 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 [2–4, 6]. 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)** [2]: $\text{ADP} = 100\% \cdot \frac{1}{N}\sum_{i=1}^{N}\frac{\max(0,Y_i^c - O_i^c)}{Y_i^c}$, 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). For ADP **lower** values indicate better results.

(2) Percentage of Increase in Confidence **(PIC)** [2]: $\text{PIC} = 100\% \cdot \frac{1}{N}\sum_{i=1}^{N}\mathbb{1}(Y_i^c < O_i^c)$. PIC reports the percentage of cases in which the model's output scores increase due to 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, in PIC **higher** values indicate a better 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 [4]. 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. In perturbation tests [4], for each image, we first extract an explanation map based on the specific explanation method. Then, we gradually mask out pixels of the input image and measure the mean top-1 accuracy of the network. 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 [6] 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 [5]. 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).

2 EEA FOR CNNS

In continuation of our discussion of EEA for CNN models, we present visual comparisons in Fig. 1 of the different CNN explanation baselines using a random sample of images from Imagenet's validation set. The examples provided demonstrate the effectiveness of EEA's explanation maps in accurately identifying and highlighting relevant objects in the image. Conversely, Grad-CAM's (GC) explanation maps are generally larger and cover a greater area, which may explain their superior performance in the ADP and PIC tests as presented in Table 5. By capturing a larger area around the object, Grad-CAM is able to maintain more relevant information in the masked image, which aids its performance in these tests. However, it is important to note that the ADP test can be problematic, as a simple all-ones mask (that effectively leaves the image unmasked) can yield an optimal ADP value of 0. Despite Grad-CAM's success in these tests, the masks generated by this method are considerably less focused on the object, as illustrated in Fig. 1.

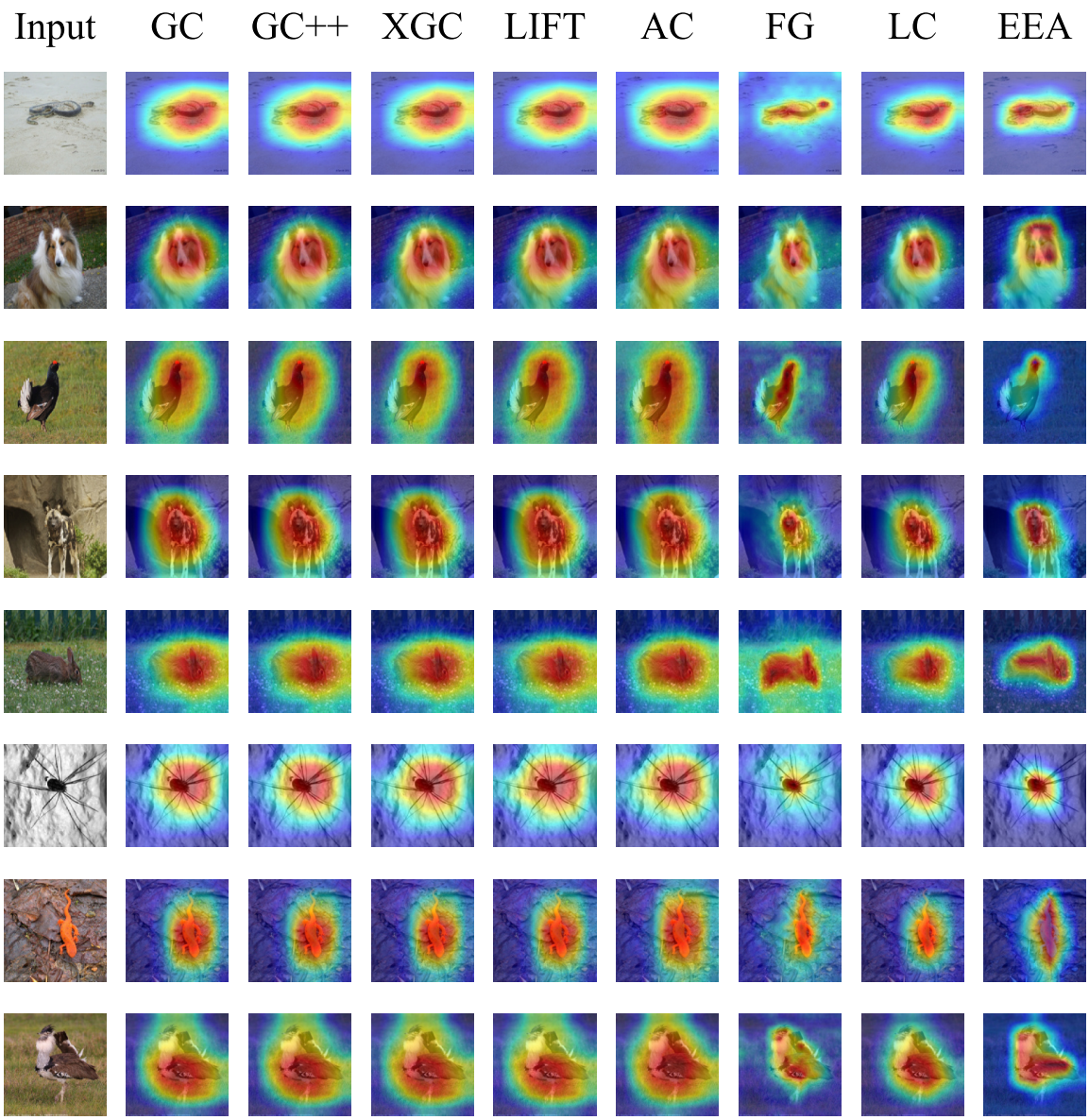

Figure 1: Sample images from ImageNet validation set for CNN models. Image classes are listed according to their row-wise appearance: "sea snake", "shetland sheepdog", "black grouse", "african hunting dog", "wood rabbit", "daddy longlegs", "eft", "bustard".

3 SANITY CHECKS FOR SALIENCY MAPS

As follow to what described in 4.6, this section expands upon the sensitivity tests of EEA. It elucidates with qualitative results to enhance the reader's understanding of sanity checks. We conducted both the *parameter randomization* and *data randomization* sanity tests as proposed by [1]. Fig. 8 demonstrates the parameters randomization and data randomization tests in the third and fourth columns, respectively, for a random sample of images (presented in the first column). We observe that the explanation maps resulting from either data or parameter randomization significantly differ from the original explanation map produced by EEA (second column). This finding indicates that EEA is sensitive to both data and parameter randomization, which is a desired property for any explanation method [1].

3.1 Parameter Randomization Test

The parameter randomization test involves a comparison of the explanation maps generated by an explanation method using two distinct configurations of the same model architecture: (1) the trained setup, where the model is trained on a specific dataset (e.g., a pretrained vit-base-patch16-224 model trained on ImageNet), and (2) the random setup, which entails the same model architecture but with random weights (e.g., a vit model initialized with random attention weights). For explanation methods that depend on the specific characteristics of the model being explained, substantial disparities are expected between the explanation maps produced for the trained model and those generated for the random model. Conversely, if the explanation maps exhibit similarity, it indicates that the explanation method is insensitive to the model's parameters, suggesting limited usefulness in terms of 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 2 presents the Spearman correlation (averaged on 50K examples) between the original explanation map obtained by EEA using the original ViT model (pretrained vit-base-patch16-224) and the explanation map obtained by EEA based cascade randomization of the original ViT model. ViT-base model uses a 12 layers of attention mechnisems. The markers on the x-axis are between '1' and '13' (total of 12 layers) where $x = k$ means that the weights of the last k layers of the model are randomized. At $x = 1$ there is no randomization, hence the correlation with the original model is perfect. Starting from $x = 2$ and up to $x = 13$, 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 EEA to the model's parameters - an expected and desired property for any explanation method [1].

Figure 3 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 EEA and the original model, while the rest of the columns present explanation maps produced by EEA 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 4 presents results for the independent randomization tests. At $x = 1$ no randomization was applied and the correlation to the original model is perfect. For $x = i$ ($i > 1$) the graph indicates the correlation of the original model with a model in which only the weights (attention scores) of the i-th layer were randomized while the weights of all other layers were kept untouched. We observe that the correlation values are low across all layers which indicates EEA'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. 5 presents a qualitative example in the same fashion as Fig. 3, 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 IIG successfully passes both types of parameter randomization tests.

3.2 Data Randomization Test

The data randomization test is a statistical method used to test the sensitivity of the explanation method to the labeling of the training data. It is carried out by producing two explanation maps using an identical architecture but with two different datasets: one with the original labels and another with randomly permuted labels. A desired result for this experiment is obtained when the model proves to be sensitive to the labeling of the dataset i.e., the produced explanation maps differ significantly between the two cases. However, if the method is insensitive to the permuted labels, it indicates that the model 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 a ViT 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 6 presents a box plot computed for the Spearman correlation values obtained for paired explanation maps (50K examples): one

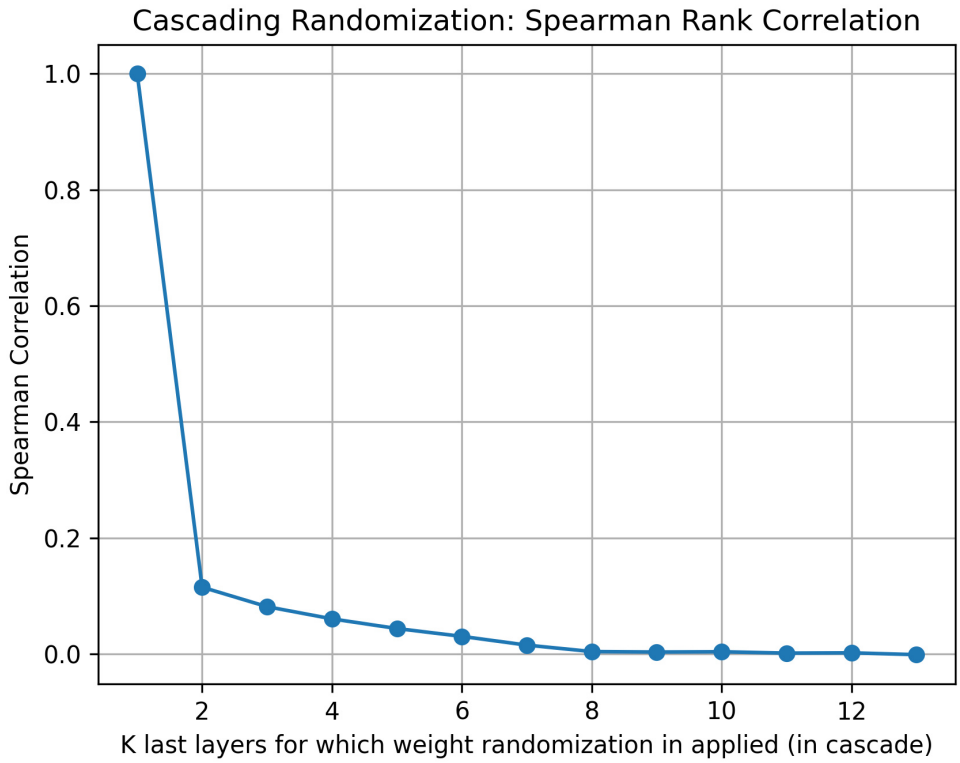

Figure 2: Cascading Randomization: The ViT model (trained on ImageNet dataset) is subjected to successive weights randomization (attention scores), beginning from the last model's layers on the. The presented graph depicts the Spearman rank correlation (averaged on 50K examples) between the explanation produced by EEA 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 first dot (x=1) corresponds to no randomization (the original model is used), hence the correlation between the explanation maps is perfect. See Sec. 3.1 for further details.

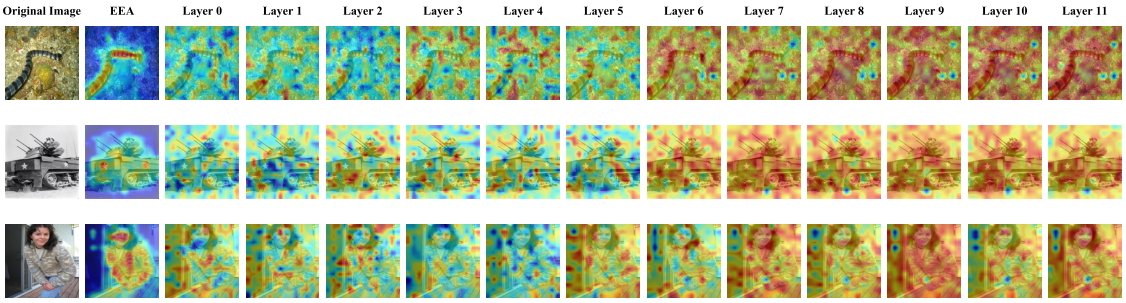

Figure 3: Cascading Randomization on EEA based on random weighted ViT model: The figure presents an original explanations progression from left to right depicts the gradual randomization of network weights up to the layer number depicted at the top of the column (starting from the last layer). See Sec. 3.1 for further details. Image classes according to their row-wise appearance: "sea snake", "half-track", "cardigan".

produced using the original model that is trained with the ground truth, and another produced by the model trained with the permuted labels. Figure 7 compares visually EEA using the original labels (column 2) and using the randomly permuted labels (column 3). We can

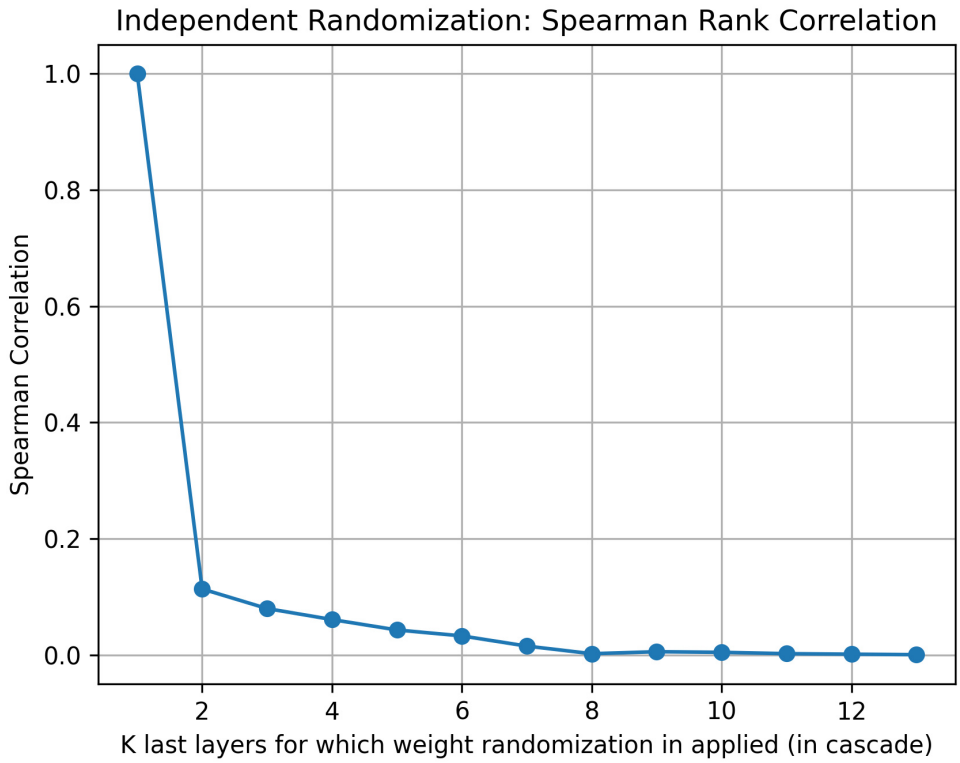

Figure 4: Independent Randomization: The randomization process is carried out independently for each layer of the model, while the remaining weights are retained at their pretrained values. The y-axis of the presented graph represents the rank correlation between the original and randomized explanations, with each point on the x-axis corresponding to a specific layer of the model.

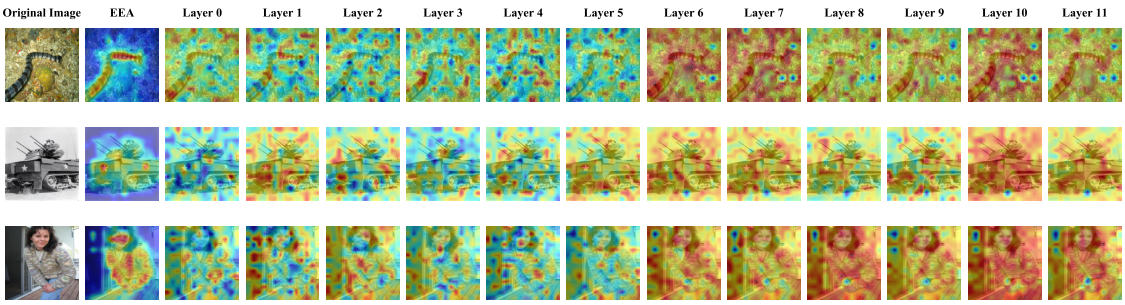

Figure 5: Independent Randomization on EEA based on random weighted ViT model: Similar to Fig. 3, however, this time, each specific layer is randomized independently, while the rest of the weights are kept at their pretrained values. Image classes according to their row-wise appearance: "sea snake", "half-track", "cardigan".

see that the correlation values are very low indicating EEA's sensitivity to the labeling of the training data. Hence, we conclude that EEA successfully passes the data randomization test.

Finally, Figure 8 provides supplementary qualitative illustrations for both tests using a pretrained vit-base-patch16-224 model. The EEA column presents the EEA map which are coherent explanation maps focused on key regions for the classification task. In contrast, in

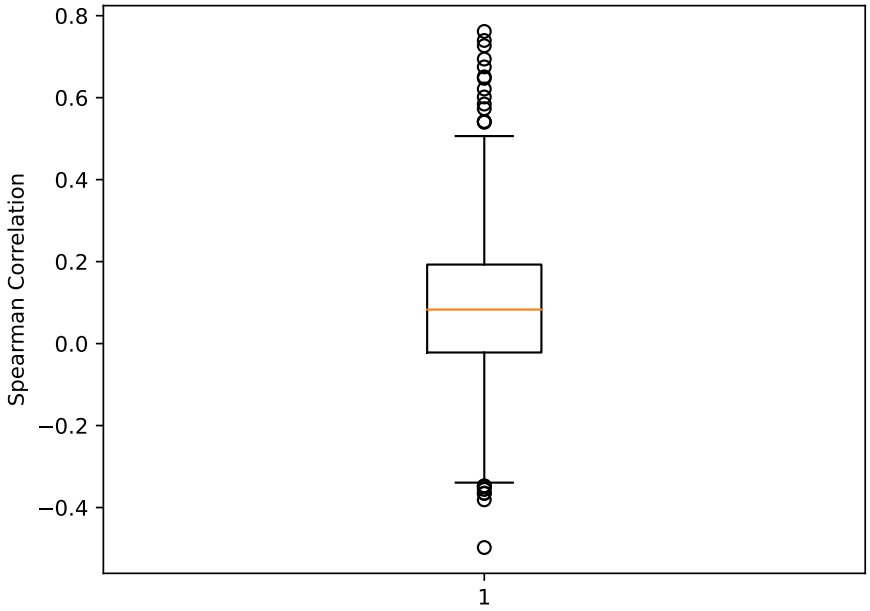

Figure 6: Data Randomization Test: Spearman rank correlation box plot for EEA with the ViT model.

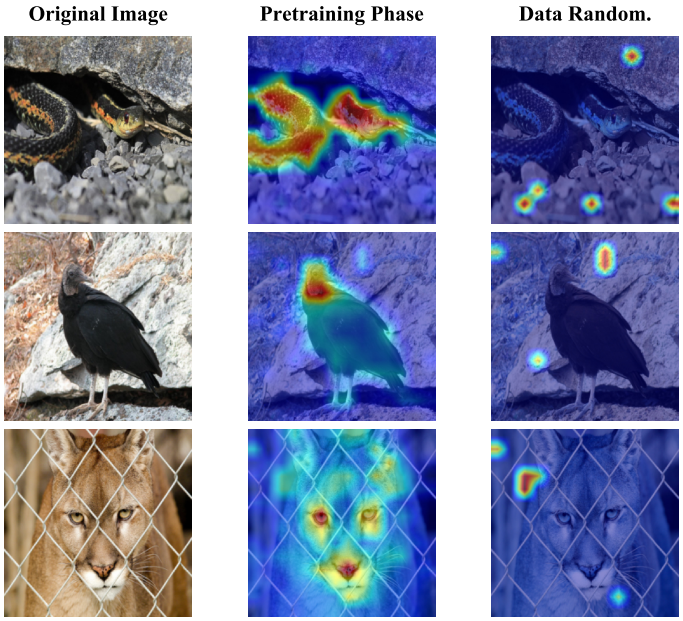

Figure 7: Data Randomization Test: A comparison between original image, EEA and EEA using a random labeled model. Top-to-bottom: "grass snake", "couga", "vulture".

the last two columns, we see the results following parameter randomization and data randomization. As can be seen, after each type of randomization, the explanation maps become irrelevant.

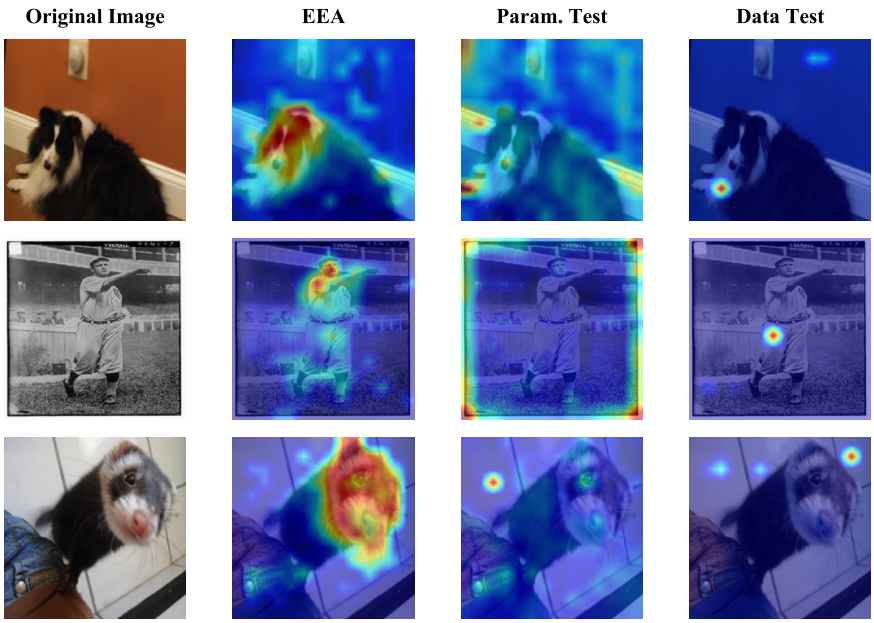

Figure 8: Sanity checks exemplified. Columns 3 and 4 present the explanation maps produced after *parameter randomization* and *data randomization*, respectively. See Sec. 3 for further details. Image classes according to their row-wise appearance: "Shetland sheepdog", "ballplayer", "Mustela putorius".

4 EEA WITH MIXED ARCHITECTURES

In our paper, we choose to implement the explainer using the same architecture as the explained model. Choosing the same architecture, and starting from pretrained weights, simplifies the learning task of the explainer. However, the EEA framework is very general and agnostic to the explainer's architecture. In order to highlight this point, in Fig 9 we present qualitative examples for different combinations of explainer-explained architectures, using ViT-B and ResNet. As can be seen, even with mixed architectures the EEA manages to produce meaningful explanation maps that capture the object of interest in all setups. This showcases the generic nature of the EEA approach which can be utilized using different architectures. As can be seen, it is possible for the explainer model to utilize a different architecture from that of the explained model.

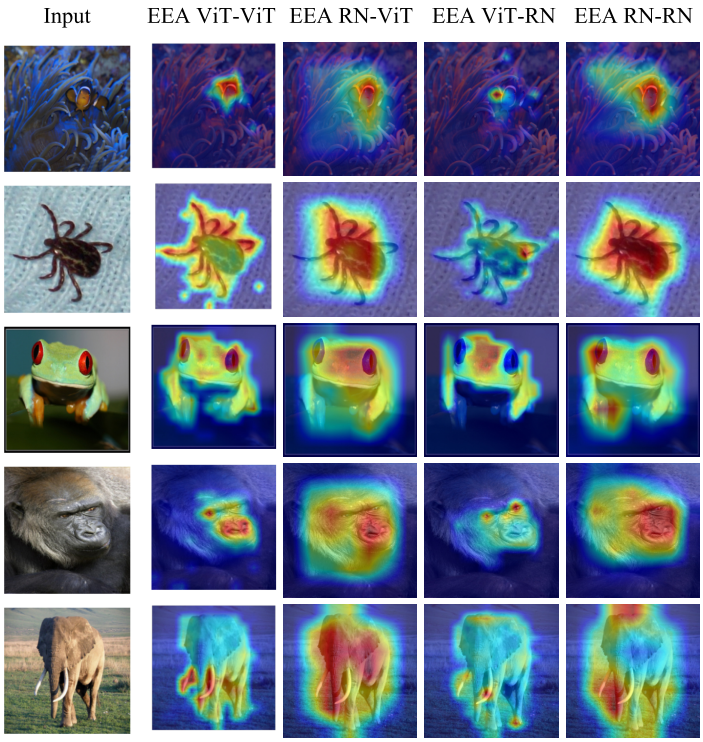

Figure 9: Qualitative examples for all possible combinations of explainer-explained architectures, using ViT-B and ResNet.

5 ADDITIONAL EXAMPLES - MULTIPLE-CLASS IMAGES

Figure 10 presents additional examples for images that combine two classes of interest. We can see that EEA provides the most accurate class-specific explanation maps.

6 ADDITIONAL EXAMPLES - SINGLE-CLASS IMAGES

Figures 11-16 present further comparative examples for explanation maps produced by EEA and the other explanation methods. These images further demonstrate EEA's advantage over its alternatives.

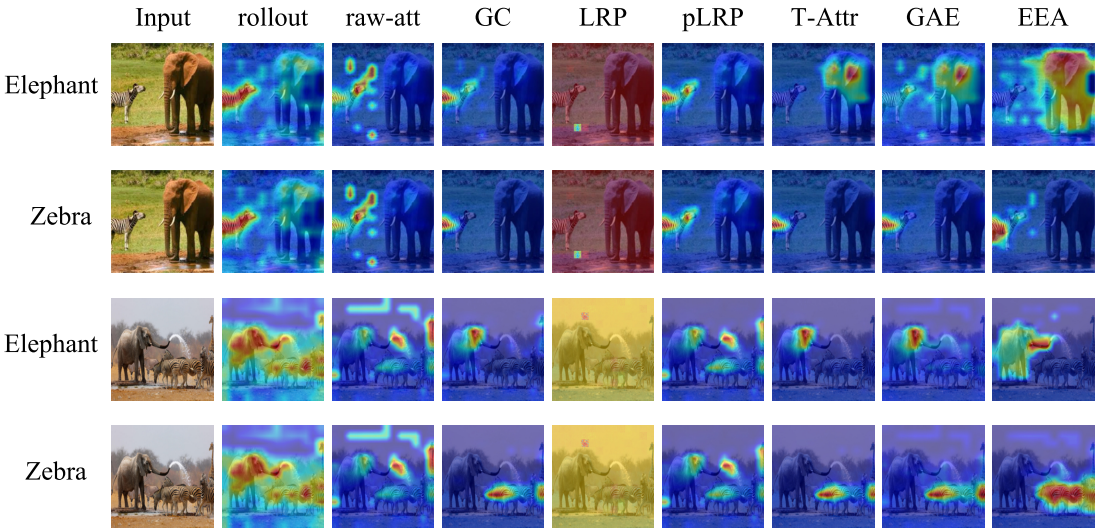

Figure 10: Class-specific visualizations for ViT models. Only GC, T-Attr, GAE, and EEA (this paper) produce class-specific maps, and EEA captures the objects most accurately.

REFERENCES

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. 2018. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*. 9505–9515.
[2] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. 2018. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*. IEEE, 839–847.
[3] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. 2018. Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. 839–847. https://doi.org/10.1109/WACV.2018.00097
[4] Hila Chefer, Shir Gur, and Lior Wolf. 2021. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 782–791.
[5] Piotr Dabkowski and Yarin Gal. 2017. Real time image saliency for black box classifiers. In *Advances in Neural Information Processing Systems*. 6970–6979.
[6] Vitali Petsiuk, Abir Das, and Kate Saenko. 2018. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421* (2018).

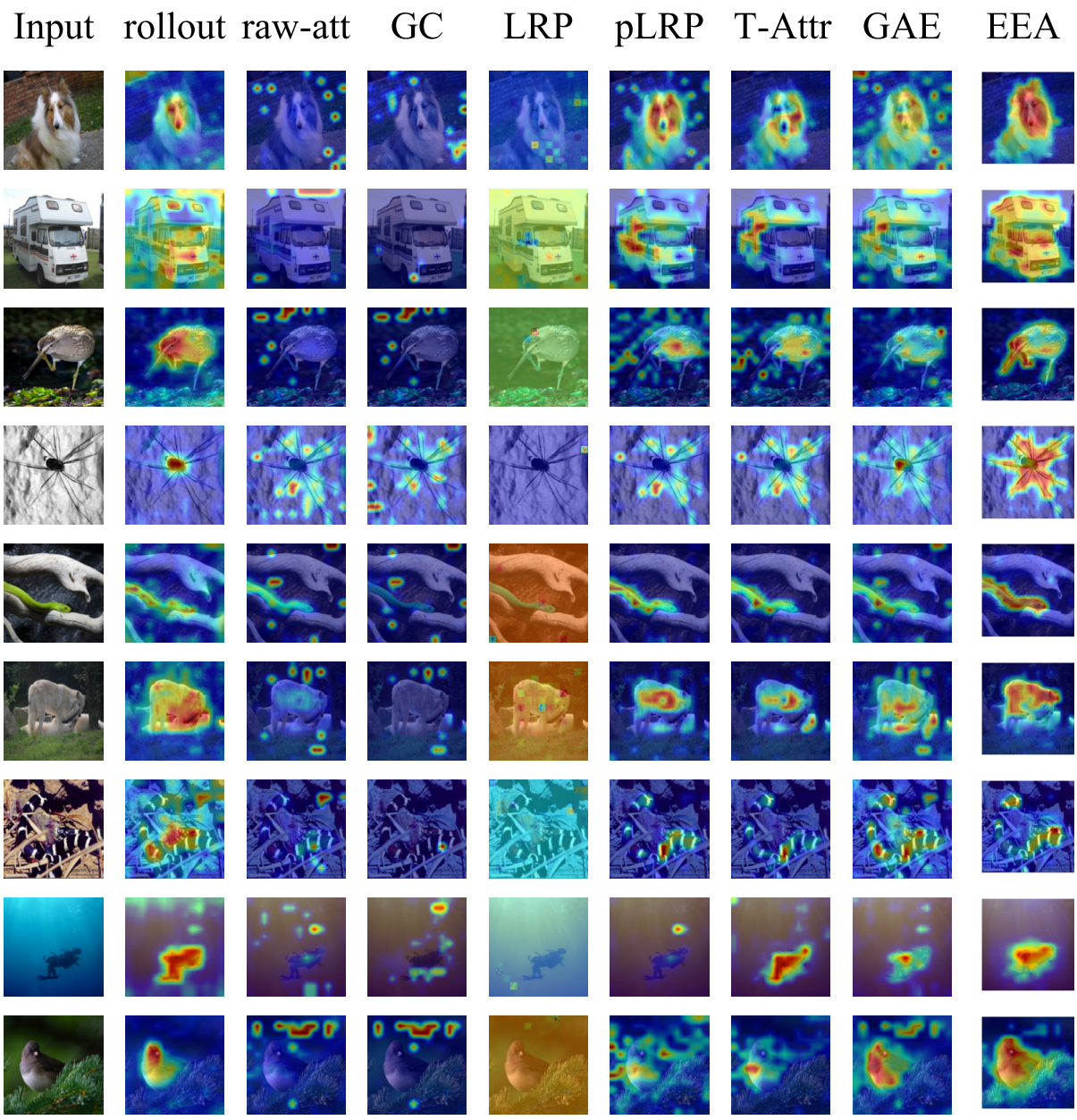

Figure 11: Sample images from ImageNet validation set for ViT models. Image classes are listed according to their row-wise appearance: "shetland sheepdog", "RV.", "dowitcher", "harvestman", "green mamba", "white wolf", "king snake", "scuba diver", "junco".

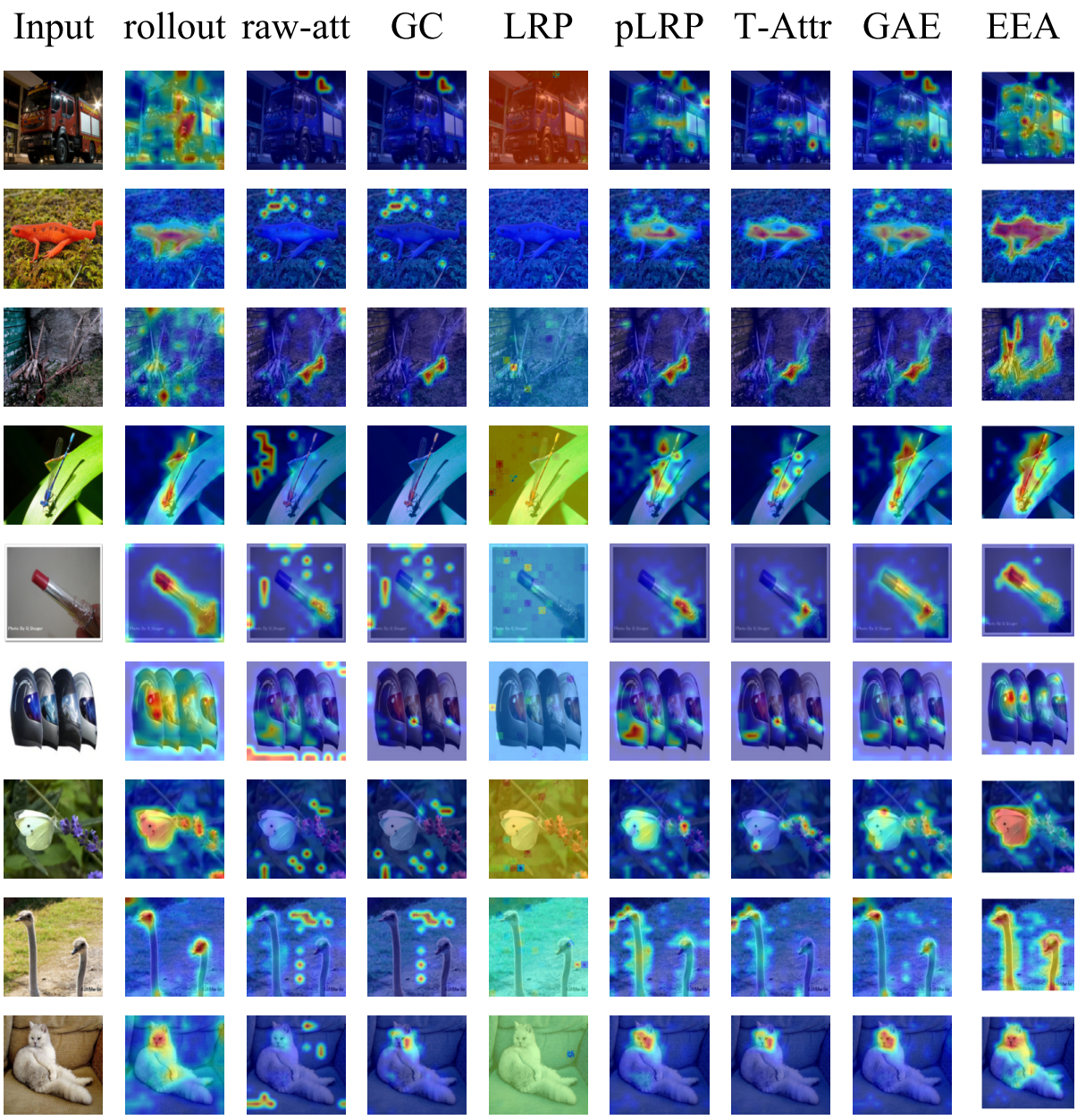

Figure 12: Sample images from ImageNet validation set for ViT models. Image classes are listed according to their row-wise appearance: "fire truck", "eft", "plow", "damselfly", "lipstick", "crash helmet", "cabbage butterfly", "ostrich", "Persian cat".

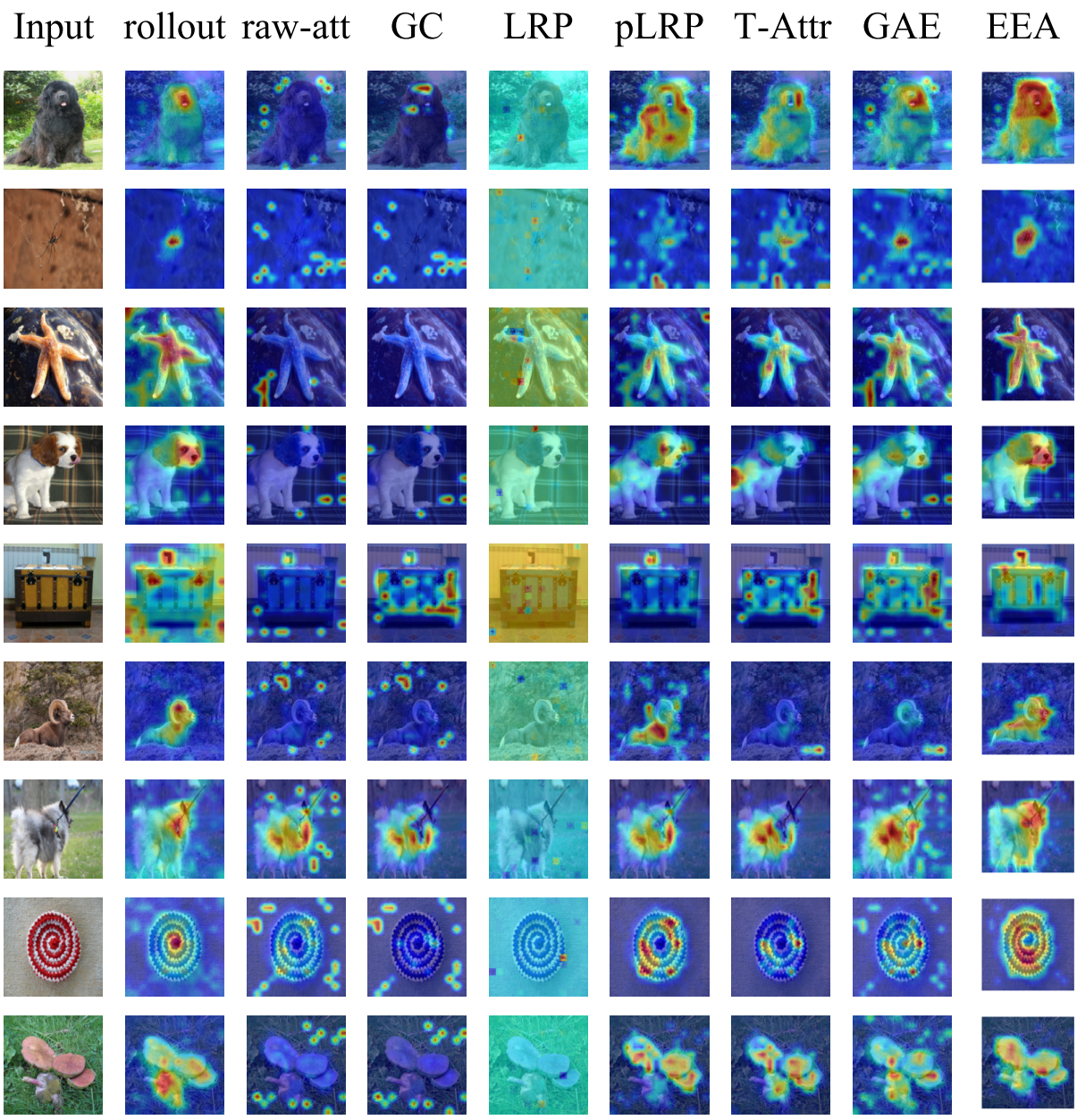

Figure 13: Sample images from ImageNet validation set for ViT models. Image classes are listed according to their row-wise appearance: "newfoundland dog", "black widow", "starfish", "blenheim spaniel", "chest", "rocky mountain sheep", "keeshond", "spiral", "bolete".

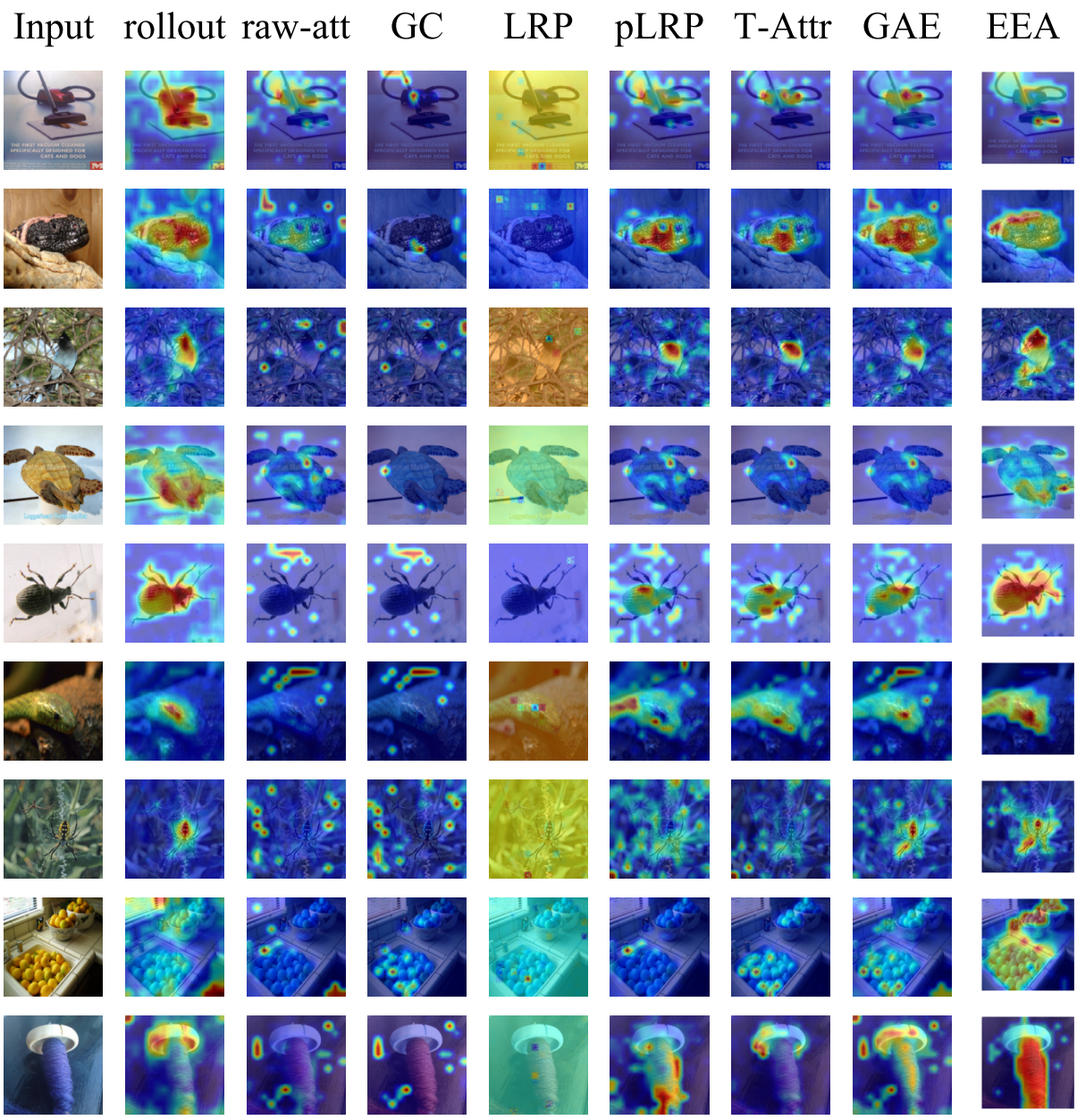

Figure 14: Sample images from ImageNet validation set for ViT models. Image classes are listed according to their row-wise appearance: "vacuum cleaner", "heloderma suspectum", "jay", "loggerhead turtle", "weevil'f", "green mamba", "black and gold garden spider", "lemon", "spindle".

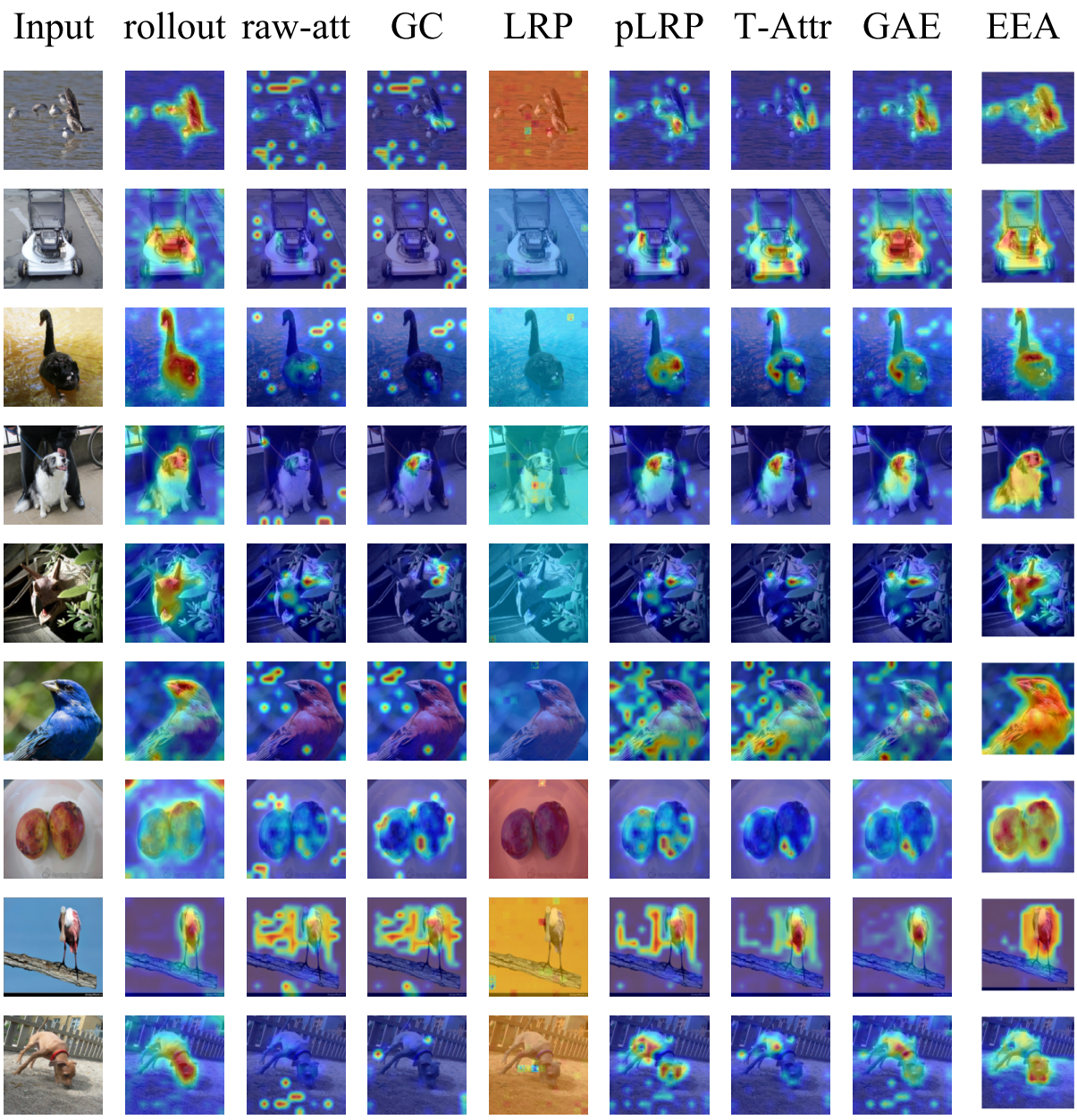

Figure 15: Sample images from ImageNet validation set for ViT models. Image classes are listed according to their row-wise appearance: "dowitcher", "lawn mower", "black swan", "border collie", "triceratops", "indigo bird", "fig", "spoonbill", "italian greyhound".

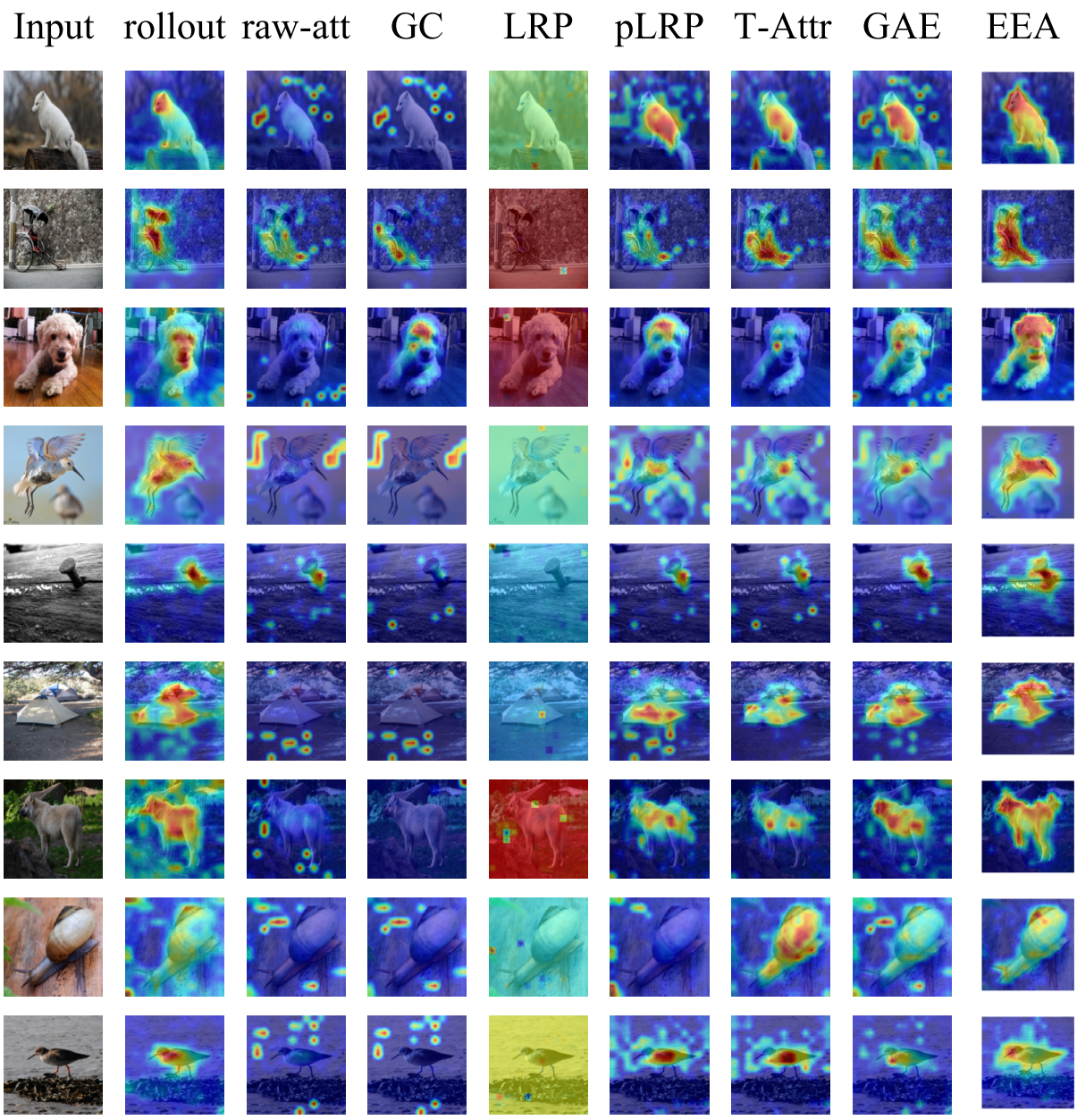

Figure 16: Sample images from ImageNet validation set for ViT models. Image classes are listed according to their row-wise appearance: "arctic fox", "ricksha", "miniature poodle", "red-backed sandpiper", "nail", "mountain tent", "white wolf", "snail", "redshank".