Supplementary Material
CAMERAS: Enhanced Resolution And Sanity preserving Class Activation Mapping for image Saliency

A-1: Computing time

Table 1. Average computational time for saliency map generation for RISE [20], Extremal Perturbation (Extremal) [8], NormGrad [21], GradCAM [25] and the proposed CAMERAS. GradCAM is the fastest. CAMERAS is generally orders of magnitude faster than the remaining methods.

Model	***RISE***	***Extremal***	***NormGrad***	***GradCAM***	***CAMERAS***
ResNet	26.4s	35.8s	7.0s	0.07s	0.44s
DenseNet	79.7s	64.1s	2.5s	0.08s	0.57s
Inception	55.9s	60.9s	5.0s	0.08s	0.54s

We compare computational time of the proposed CAMERAS with other techniques in Table 1 that reports the average processing time of an image, estimated over 1000 images for three different visual models. For a fair comparison, all of the experiments are performed on NVIDIA Titan V GPU with Pytorch framework, keeping the batch size 1. Publicly available implementations are used for NormGrad [21], Extremal Perturbations [8] and RISE [20]. For GradCAM [25], we use the Torchray package [8]. For the Extremal perturbations, we fix the hyper-parameters 'area' to 30% and iterations to 1000. Similarly, RISE has a fixed maximum number of 6000 iterations. These settings generally achieve visually comparable results to CAMERAS.

Table 1 shows that CAMERAS computational time is slightly higher than Grad-CAM, which is a highly efficient method for saliency computation. This happens because CAMERAS requires multiple forward/backward passes to construct the saliency map. However, the overall computational cost resulting from these passes are far less than the other methods. This is reflected in the order of magnitudes higher processing times required by RISE and Extremal perturbations.

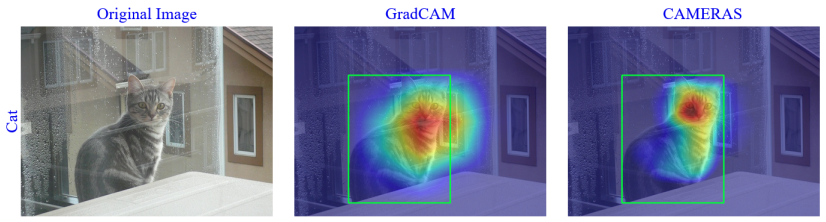
A-2: CAMERAS parameters

Table 2. Average difference between CAMERAS saliency maps generated with different resolution steps ('N') and fixed maximum resolution $\zeta_m = (1K, 1K)$. The results were computed for ResNet-50 over 1000 randomly sampled images from ILSVRC 2012 validation dataset.

Steps $\downarrow\rightarrow$	$N=5$	$N=6$	$N=7$	$N=8$	$N=9$	$N=10$
$N=5$	0.00	0.11	0.09	0.11	0.10	0.11
$N=6$	0.11	0.00	0.13	0.08	0.15	0.10
$N=7$	0.09	0.13	0.00	0.11	0.07	0.09
$N=8$	0.11	0.08	0.11	0.00	0.12	0.07
$N=9$	0.10	0.15	0.07	0.12	0.00	0.10
$N=10$	0.11	0.10	0.09	0.07	0.10	0.00

CAMERAS accepts resolution steps ('N') as a hyper-parameter. The performance of our technique remains largely insensitive for $N \in [5, 10]$. In order to demonstrate that, we report the average difference between the saliency maps of our method for different values of N. A small difference signifies only slight variations in the maps and hence insensitivity of CAMERAS to the hyper-parameter. In Table 2, we report the average difference values, where the value for a step size between N_1 and N_2 is computed as:

$$\mathrm{E}\Big[\frac{||(\Psi_1 - \Psi_2)||_2}{(||\Psi_1||_2 + ||\Psi_2||_2)/2}\Big] \;\; \text{s.t} \;\; \Psi_1 = \text{CAMERAS}(N_1), \Psi_2 = \text{CAMERAS}(N_2), \tag{1}$$



Pointing Game Score	1.0	1.0
$\rho^{+}_{map}/\rho^{-}_{map}$	2.9/0.3	5.4/0.3

Figure 1. Saliency maps for an image from Pascal VOC 2007 [7] over ResNet-50 model. The second column includes the result for Grad-CAM (top-row), its score for the pointing game (middle row) and its score on our proposed metrics score (last row). The corresponding results for CAMERAS are given in the last column. Our metrics account for the overall precision on the saliency map.

From the table 2, it can be observed that the difference between the saliency maps stays within 15% of the average norm of the original maps. This is true even with the reduction in number of steps by half. This validates that the performance of CAMERAS is reasonably stable for different values of 'N'.

A-3: Further discussion on the proposed evaluation metrics

In the main paper, we indicated that due to the crudeness of the saliency maps resulting from the earlier methods, existing evaluation metrics are also meant to evaluate the maps imprecisely. For instance, Pointing game [34] evaluates a saliency map by checking if the maximal point in the saliency map lies with in the bounding box of an object. If this is true, it assigns the map the best score. However, no attention is paid to the rest of the region identified by the map. For example, in Figure 1 (top-row) Pointing game assigns equal score to the saliency maps of 'GradCAM' ($\Psi_{GradCAM}$) and 'CAMERAS' ($\Psi_{CAMERAS}$), ignoring the fact that Grad-CAM map also includes the background region outside the bounding box. The binary nature of the score does not reveal much about the actual quality of the attribution maps. Therefore, a more precise metric is needed that scores the attribution maps corresponding to model confidence over the salient region. Our proposed metrics capture this notion adequately. This is reflected in Figure 1 where Grad-CAM clearly gets penalized for inclusion of the irrelevant pixels under our evaluation metrics.

We further scrutinize the behaviour of the proposed metrics over the inclusion of irrelevant pixels. Inclusion of irrelevant pixels is generally the result of blobiness caused by interpolation in the existing methods. This imprecision should result in reduction of an appropriate metric score. We empirically verify this for our metrics by linearly increasing the CAMERAS saliency map to the GradCAM map. This is performed by setting different values of γ in Equation 2

$$\Psi_{interpolated} = \Psi_{CAMERAS} + \gamma(\Psi_{GradCAM} - \Psi_{CAMERAS}) \;\; \text{s.t} \;\; \gamma \in [0, 1]. \tag{2}$$

We gather a number of interpolated saliency maps by incremental increase in the salient region. The proposed metric is computed over these interpolated saliency maps and their trend is analyzed. Figure 2 shows some of the interpolated saliency maps (left) and reports the metric variation in a plot (right). The graph matches our intuition that as more and more irrelevant pixels are included, the scores decreases monotonically. This demonstrates that the metric adequately captures the attribution from all the pixels. We note that, for all of these interpolated saliency maps, the pointing game score stays the same. This reinforces our argument for the need of more precise evaluation metrics for image saliency.

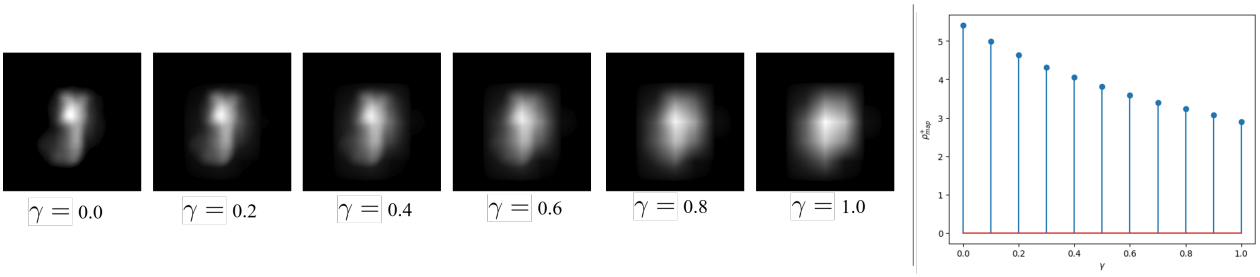


Figure 2. Interpolated saliency maps computed under Equation 2 with changing value of γ. For $\gamma = 0$, the map corresponds to CAMERAS. For $\gamma = 1$, the map resembles to that computed by GradCAM. The graph on the right shows the corresponding impact on ρ^{+}_{map} score.

A-4: Adversarial attacks inspired sanity check

Adversarial attacks alter image pixels to change the prediction of a model. These alterations are systematic - in the directions that maximally alter the model output. Intuitively, cleaning the adversarially attacked pixels by replacing them with the original pixels, should restore the model's confidence on the original label of the image. This observation inspires a simple sanity check for the image saliency methods - a test that implicitly accounts for the precision of the map.

Saliency methods ultimately aim at tallying the model-centric importance of individual pixels of an image. Therefore, for a 'sane' method that assigns correct importance to the individual pixels, iteratively cleaning the adversarial version of those pixels in a sorted manner should result in a steep rise of model's confidence on the original image label. For a saliency map that fails to compute the correct importance for the individual pixels, restoration of the confidence with this process is expected to be slow and unstable. This behavior indicates the method to be 'less' sane. We make two observations here. First, our sanity check does not have a binary outcome. Though less acknowledged, this fact also holds for the other popular sanity checks. For instance, the model layer randomisation of [2] also does not offer a binary decision. Thus, a method can be less sane than another under the proposed sanity check. Second, our definition of sanity is based on individual pixel importance. This is a tough criterion, emulating the hardest possible scenario for the perfect sanity. That is, only the method that computes correct relative importance for all the pixels in an image can perfectly pass this test.

To observe a map's sanity, we suggest iterative cleaning of the adversarial pixels in an attacked version of the image in the descending order of pixel importance, as stipulated by the computed saliency map. An ideal saliency map will raise the ground truth confidence of the model almost instantly under this cleaning process. Therefore, the area under probability-pixels curve for an ideal saliency map will be

$$Area_{\text{Ideal}} \approx P_{\text{GroundTruth}} \times H \times W \;\; \text{s.t} \;\; P_{\text{GroundTruth}} = P\big(\mathcal{K}(I)\big),\; I \in \mathbb{R}^{H \times W \times C}, \tag{3}$$

Any deviation from the ideal saliency mask will introduce a corresponding change in the area under the curve, which can be used as an indicator of sanity. The deviation can be computed by observing the difference in the area under probability-pixels graph of an ideal and a given saliency map. A smaller difference will be more desirable.

To implement this sanity check, we corrupt a given image by the proposed enhanced PGD scheme and afterward iteratively clean the resulting adversarial image in the order of importance indicated by the saliency map. The computational complexity of this process is directly proportional to number of pixels involved. Therefore, we fix the maximum number of pixels to be cleaned to a fraction of the original image size. We evaluate the sanity score as,

$$Sanity = \frac{Area_{\text{Ideal}} - Area_{\Psi}}{Area_{\text{Ideal}}}, \tag{4}$$

where '$Area_{Ideal}$' indicates area of the ideal scheme and '$Area_{\Psi}$' indicates the area of a given image saliency method. This formulation allows to rank the sanity as a real number in $[0, 1]$. Figure 3 shows the sanity of Grad-CAM saliency maps and CAMERAS for a number of images. It highlights that CAMERAS saliency maps result in steep probability-pixel curves than those of Grad-CAM. This is reflected in their respective sanity scores indicated in the yellow fonts. The low score validates that CAMERAS saliency maps are much more sane than those of Grad-CAM.

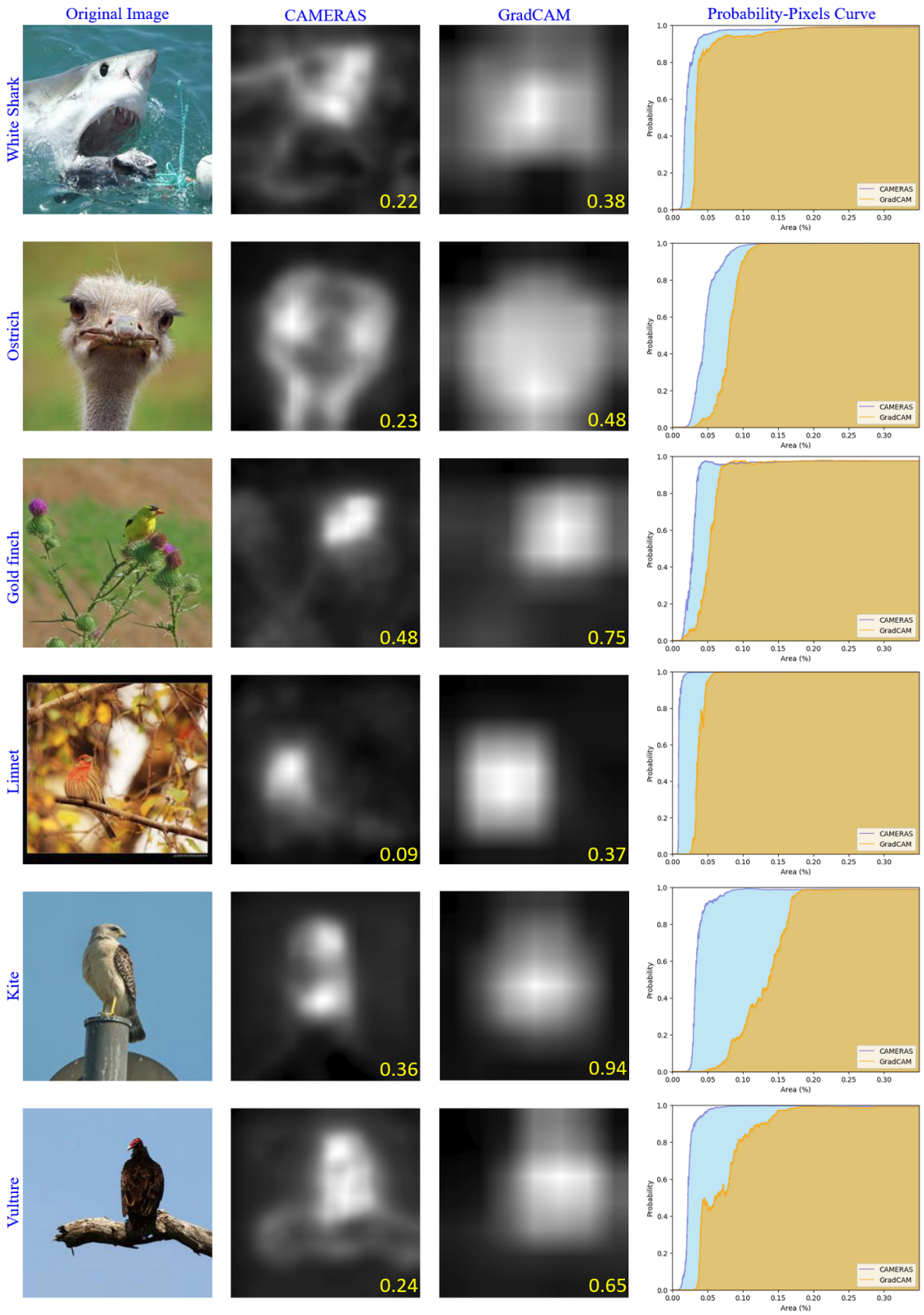


Figure 3. Probability-pixels curves (last column) of Grad-CAM (Brown) and CAMERAS (Blue) for the image shown in first column. The saliency map from CAMERAS is shown in second column and Grad-CAM in the third column respectively. The sanity scores (smaller is better) are indicated over the saliency maps in 'yellow' fonts. Scores are evaluated with the ideal area composed of 10% image pixels.

A-5: Further Qualitative Comparison Results

We have included additional results for ResNet, DenseNet and Inception in Figure 4, 5 and 6 respectively.

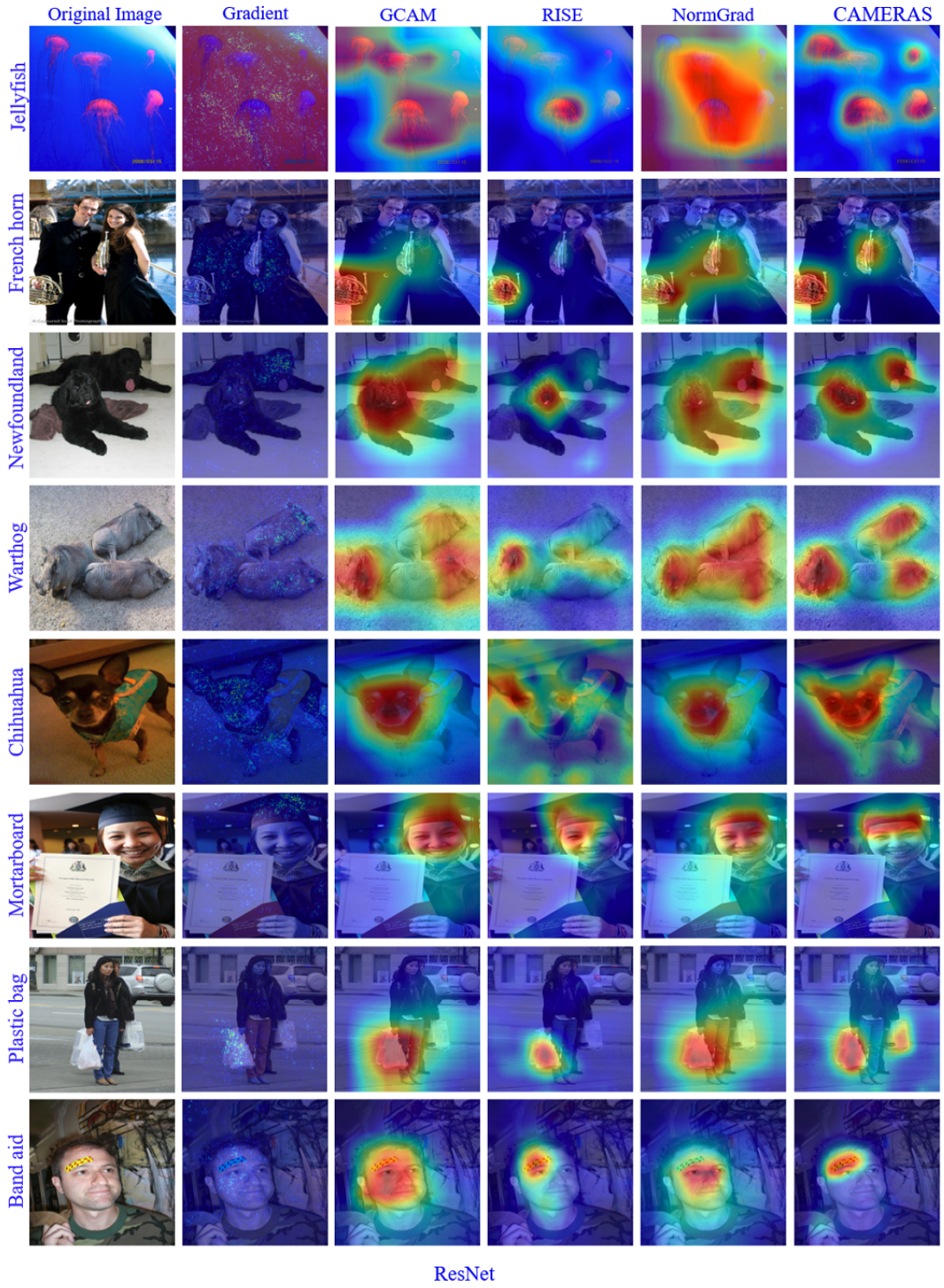


Figure 4. Qualitatively comparison of CAMERAS attribution maps with other schemes including Gradient [27], Grad-CAM (GCAM), [25], RISE [20] and NormGrad[21]. All of the results are generated with ImageNet pretrained ResNet-50.

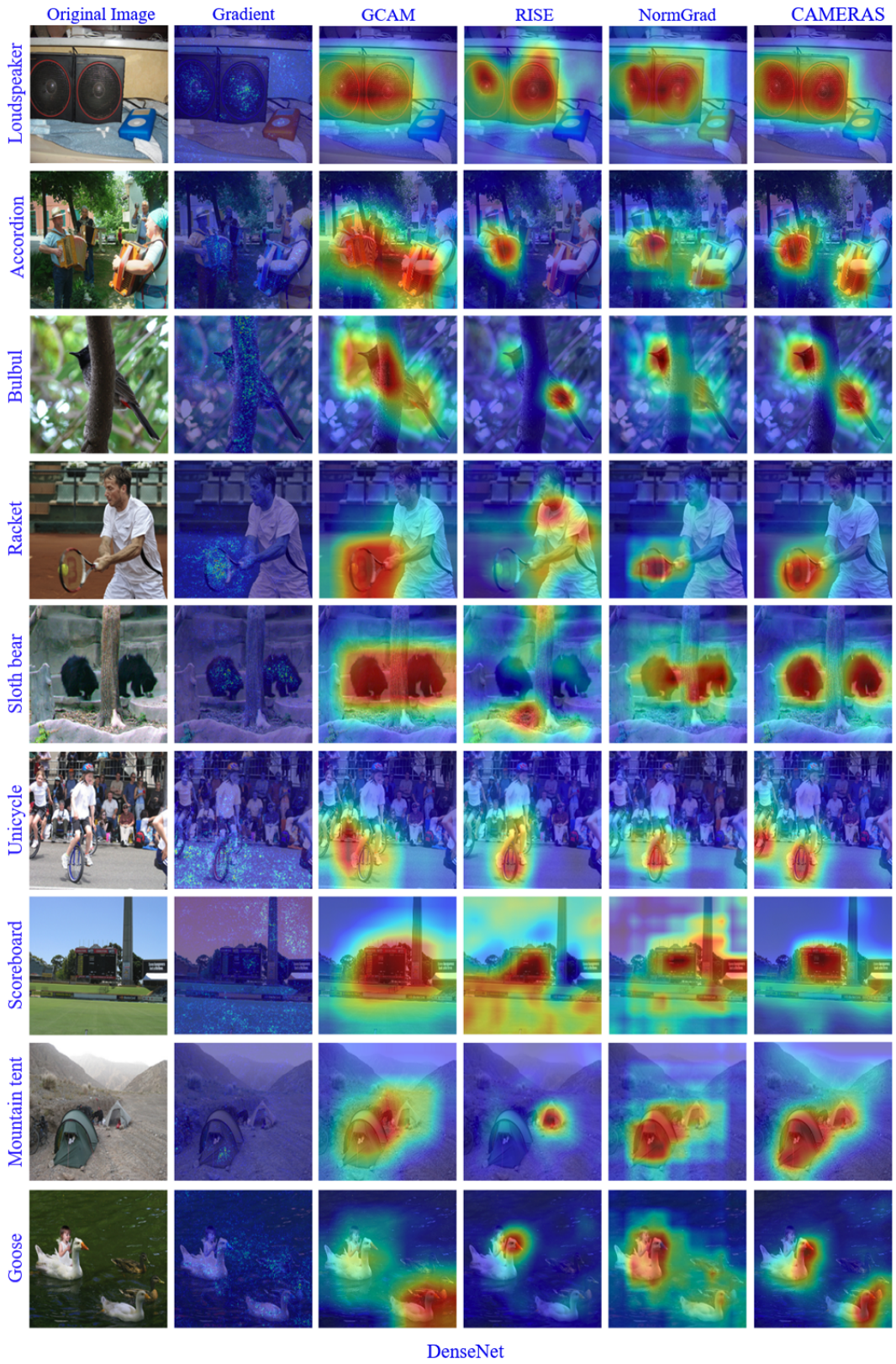


Figure 5. Qualitatively comparison of CAMERAS attribution maps with other schemes including Gradient [27], Grad-CAM (GCAM), [25], RISE [20] and NormGrad[21]. All results are generated with ImageNet pretrained DenseNet-121.

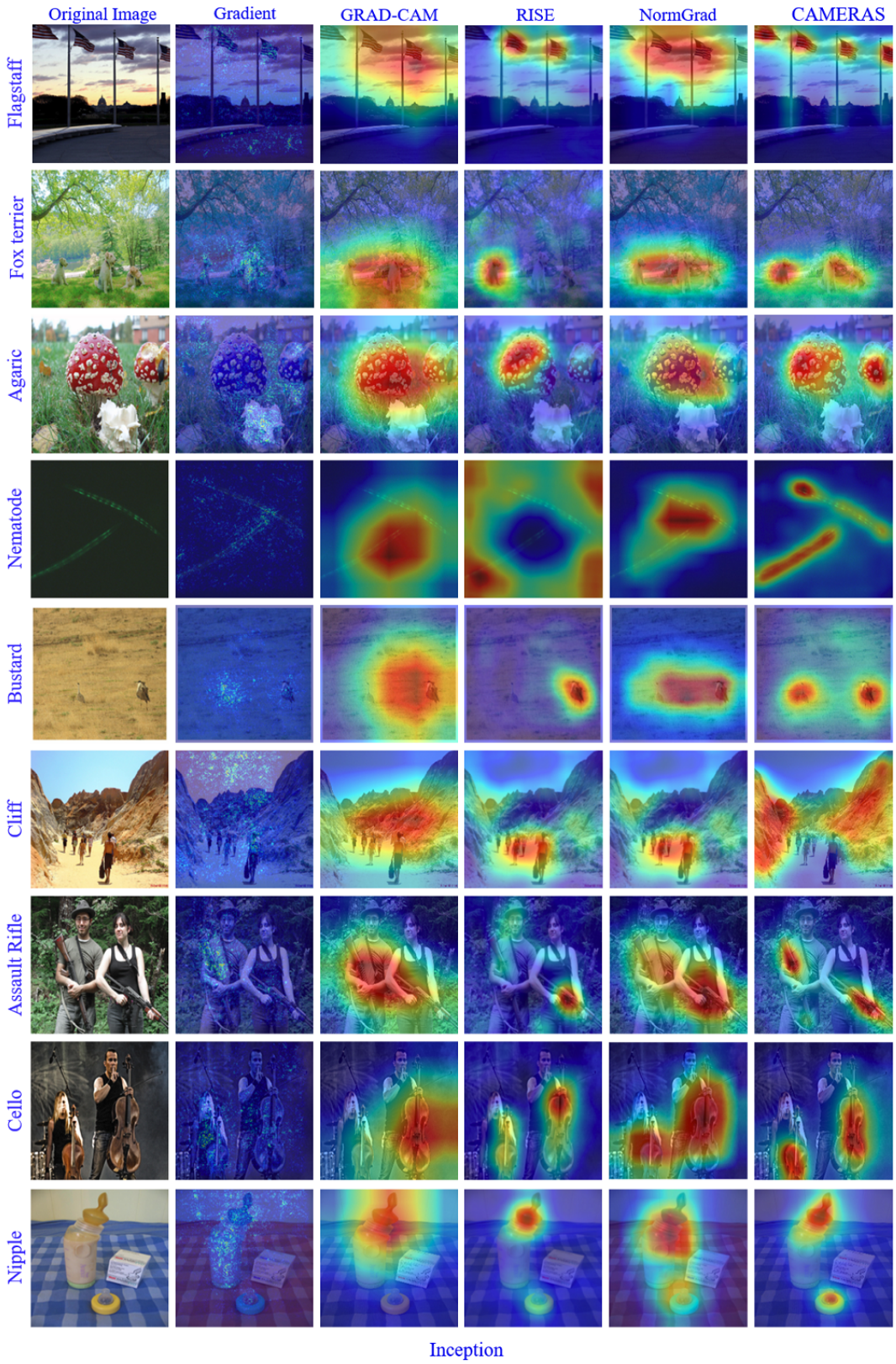


Figure 6. Qualitatively comparison of CAMERAS attribution maps with other schemes including Gradient [27], Grad-CAM (GCAM), [25], RISE [20] and NormGrad[21]. All results are generated with ImageNet pretrained Inception-V3.

A-6: Further results for Enhanced PGD

We have included further results of Enhanced PGD in Figure 7. These results highlight that CAMERAS saliency maps effectively enhance the visual quality of perturbations while maintaining similar fooling confidence.

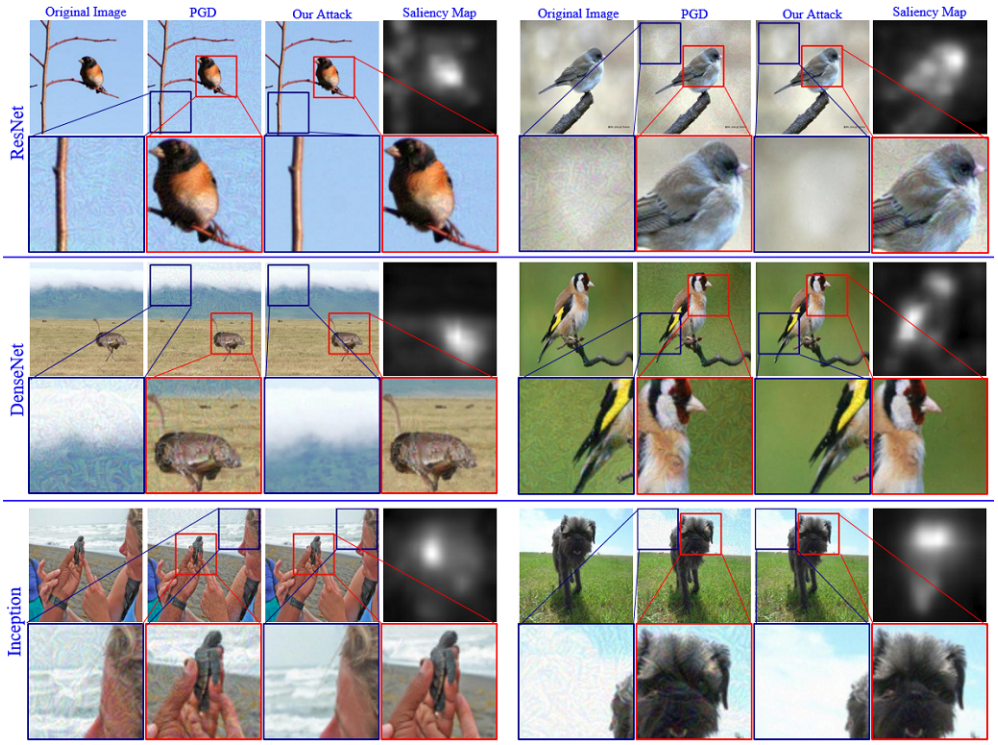


Figure 7. Comparison of visual quality of adversarial images with Vanilla PGD [18] and Enhanced PGD for three different visual classifiers. These attacks were generated with $\epsilon = 12/255$, maintaining 99.99% confidence on incorrect classes.

A-6: Further results for Prediction Confidence

Additional results showing similarities in the attention of different models on similar features to achieve similar confidence are given in Figure 8.

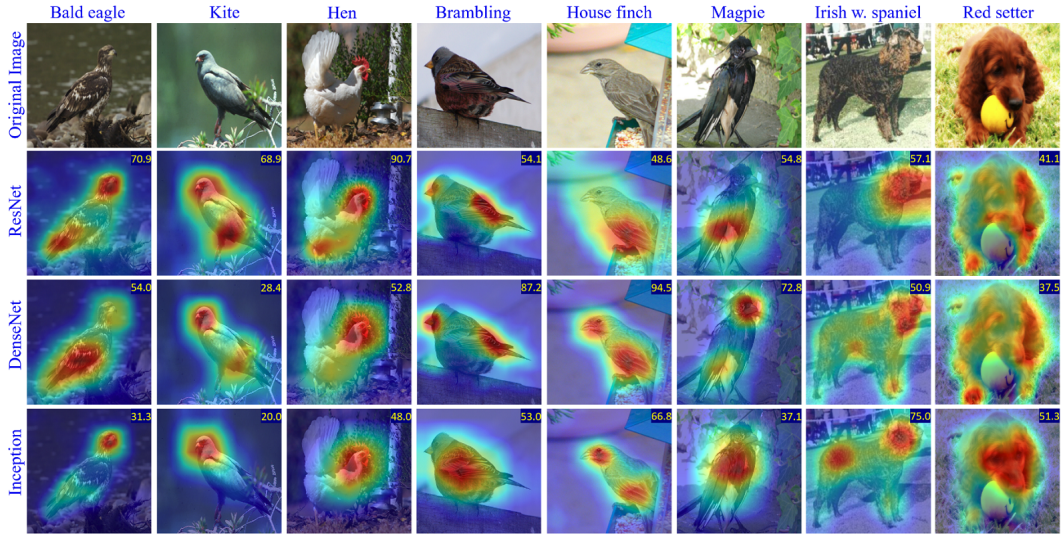


Figure 8. Precise saliency of CAMERAS reveals similarity in the level of attention on fine-grained features causes similarity in the prediction confidence (given as percentages) of different models.