Appendix A. Incorporating Bias Terms

In the following, we provide a simple example to show that the output for a ReLU neural network with bias terms can be computed as in Eq. (13). In particular, let us consider the output for a network with 3 hidden layers:

$$y = \mathbf{w}^T \phi\Bigg(\mathbf{W}_3\Big[\phi(\mathbf{W}_2\big[\phi(\mathbf{W}_1\mathbf{x}+\mathbf{b}_1)\big]+\mathbf{b}_2)\Big]\Bigg) \tag{17}$$

For the sake of simplicity and without loss of generality, we assume that no bias is applied to the last hidden layer. Given the defined network, the output y can be obtained by the sequential computation of the intermediate layers as:

$$\begin{aligned}
\mathbf{h}_1 &= \mathbf{W}_1\mathbf{x} \\
\mathbf{h}_2 &= \mathbf{W}_2\phi(\mathbf{h}_1+\mathbf{b}_1) = \hat{\mathbf{W}}_2\mathbf{W}_1\mathbf{x} + \hat{\mathbf{W}}_2\mathbf{b}_1 \\
\mathbf{h}_3 &= \mathbf{W}_3\phi(\mathbf{h}_2+\mathbf{b}_2) = \hat{\mathbf{W}}_3\hat{\mathbf{W}}_2\mathbf{W}_1\mathbf{x} + \hat{\mathbf{W}}_3\hat{\mathbf{W}}_2\mathbf{b}_1 + \hat{\mathbf{W}}_3\mathbf{b}_2 \\
y &= \mathbf{w}^T\phi(\mathbf{h}_3) = \hat{\mathbf{w}}^T\hat{\mathbf{W}}_3\hat{\mathbf{W}}_2\mathbf{W}_1\mathbf{x} + \hat{\mathbf{w}}\hat{\mathbf{W}}_3\hat{\mathbf{W}}_2\mathbf{b}_1 + \hat{\mathbf{w}}^T\hat{\mathbf{W}}_3\mathbf{b}_2
\end{aligned} \tag{18}$$

where $\hat{\mathbf{W}}_l = \mathbf{W}_l \operatorname{diag}(\mathcal{H}(\mathbf{h}_{l-1}+\mathbf{b}_{l-1}))$, and $\hat{\mathbf{w}} = \hat{\mathbf{w}} \operatorname{diag}(\mathcal{H}(\mathbf{h}_3))$. By using the same procedure for any network with an arbitrary depth, it is clear that the network output can be expressed by using Eq. (13) in the main paper:

$$y = \hat{\mathbf{w}}^T\Big[\hat{\mathbf{W}}_L \dots \mathbf{W}_1\Big]\mathbf{x} + \sum_{l=2}^{L} \hat{\mathbf{w}}^T\Big[\hat{\mathbf{W}}_L \dots \hat{\mathbf{W}}_l\Big]\mathbf{b}_{l-1}, \tag{19}$$

Appendix B. DMBP Optimization

In Algorithm 1, we provide the pseudocode for computing attribution maps with DMBP over CNNs.

Algorithm 1 Attribution Map Compuptation with DMBP

Input: Image: $\mathbf{x}$, Network: F, Target label: y
Output: Attribution Map: $\mathbf{a}$

// Initialization
Initialize filter masks $\mathbf{\Sigma}_l$ for all l as described in Appendix B.1
// Initial Forward Pass
Compute $y = F(\mathbf{x})$
// Linearize the CNN function with Eq. (13) and minimize Eq. (10) with gradient descent
for $i = 1$ to *iters* **do**
  // Compute positive term $y(\boldsymbol{\sigma})^+$ using Eq. (11) //
  $\nabla_{\mathbf{x}}^+ F(\mathbf{x}) \leftarrow$ Backward pass with $\mathbf{\Sigma}_l \odot \nabla_{\mathbf{h}_l} F(\mathbf{x})$
  $y(\mathbf{x})^+ = \nabla_{\mathbf{x}}^+ F(\mathbf{x})^T\mathbf{x} + \sum_{l=0}^{L} \nabla_{\mathbf{b}_l}^+ F(\mathbf{x})^T\mathbf{b}_l$

  // Compute negative term $y(\boldsymbol{\sigma})^-$ using Eq. (11) //
  $\nabla_{\mathbf{x}}^- F(\mathbf{x}) \leftarrow$ Backward pass with $(1-\mathbf{\Sigma}_l) \odot \nabla_{\mathbf{h}_l} F(\mathbf{x})$
  $y(\mathbf{x})^- = \nabla_{\mathbf{x}}^- F(\mathbf{x})^T\mathbf{x} + \sum_{l=0}^{L} \nabla_{\mathbf{b}_l}^- F(\mathbf{x})^T\mathbf{b}_l$

  // Compute nuisance term $y(\mathbf{x})^\sim$ //
  $y^\sim(\boldsymbol{\sigma}) = y - y^+(\boldsymbol{\sigma}) - y^-(\boldsymbol{\sigma})$

  // Loss optimization //
  Compute $\mathcal{L} = y^-(\boldsymbol{\sigma}) - y^+(\boldsymbol{\sigma}) + \|y^\sim(\boldsymbol{\sigma})\|_1$
  Compute loss gradients $\nabla_{\mathbf{\Sigma}_l}\mathcal{L}$ for all $\mathbf{\Sigma}_l$
  Update $\mathbf{\Sigma}_l$ for all l
end for
// Return attribution map $\mathbf{a}$ *//*
$\mathbf{a} = \nabla_{\mathbf{x}}^+ F(\mathbf{x}) \odot \mathbf{x} + \nabla_{\mathbf{x}}^- F(\mathbf{x}) \odot \mathbf{x}$

B.1. Initializing filter masks $\mathbf{\Sigma}_l$

In our preliminary experiments, we observed that a random initialization of parameters $\mathbf{\Sigma}_l$ produces suboptimal results in some cases. In order to provide a better initial point for DMBP optimization, we use the following procedure.

Starting from the last layer L, we perform two parallel backward passes computing the positive and negative gradients $\nabla_{\mathbf{x}} F^+(\mathbf{x})$ and $\nabla_{\mathbf{x}} F^-(\mathbf{x})$ using $\mathbf{\Sigma}_l$ and $\mathbf{I} - \mathbf{\Sigma}_l$, respectively. During the backpropagation process, each element i of the filter masks is initialized for each layer l as:

$$\mathbf{\Sigma}_l^i = \begin{cases} \text{sigmoid}(2) & \text{iff} \quad \nabla^i_{\mathbf{h}_l} F^+(\mathbf{x}) > 0 \;\text{ and }\; \nabla^i_{\mathbf{h}_l} F^-(\mathbf{x}) > 0 \\ \text{sigmoid}(-2) & \text{iff} \quad \nabla^i_{\mathbf{h}_l} F^+(\mathbf{x}) < 0 \;\text{ and }\; \nabla^i_{\mathbf{h}_l} F^-(\mathbf{x}) < 0 \\ 0.5 & \text{otherwise} \end{cases}$$

where $\nabla_{\mathbf{h}_l} F^+(\mathbf{x})$ is the positive gradient of the output w.r.t. the activations $\mathbf{h}_l$. Similarly, $\nabla_{\mathbf{h}_l} F^-(\mathbf{x})$ is the negative gradient obtained by using $\mathbf{I} - \mathbf{\Sigma}_l$ during backpropagation.

The proposed initialization procedure is motivated by the fact that, ignoring the bias terms, the positive and negative outputs $y^+(\boldsymbol{\sigma})$ and $y^-(\boldsymbol{\sigma})$ can be expressed as:

$$y^+(\boldsymbol{\sigma}) = [\nabla_{\mathbf{h}_l} F^+(\mathbf{x}) \odot \mathbf{\Sigma}_l]^T \mathbf{h}_l \quad , \quad y^-(\boldsymbol{\sigma}) = [\nabla_{\mathbf{h}_l} F^-(\mathbf{x}) \odot (\mathbf{I} - \mathbf{\Sigma}_l)]^T \mathbf{h}_l, \tag{20}$$

where $\mathbf{h}_l \geq 0$ is the result of a ReLU operation and thus, it is always positive. As a consequence, we can locally increase the value of $y^+(\boldsymbol{\sigma})$ by assigning low values to elements $\mathbf{\Sigma}_l^i$ when $\nabla^i_{\mathbf{h}_l} F^+(\mathbf{x})$ is negative. Additionally, large values for $\mathbf{\Sigma}_l^i$ must be assigned to elements where $\nabla^i_{\mathbf{h}_l} F^+(\mathbf{x})$ is positive. Given that the negative score $y^-(\boldsymbol{\sigma})$ is computed using $\mathbf{I} - \mathbf{\Sigma}_l$ during backpropagation, our initialization procedure also takes into account the negative gradient $\nabla^i_{\mathbf{h}_l} F^-(\mathbf{x})$. In particular, we decrease $y^-(\boldsymbol{\sigma})$ by assigning a high value to $1 - \mathbf{\Sigma}_l^i$ when $\nabla^i_{\mathbf{h}_l} F^-(\mathbf{x})$ is negative. Similarly, we set a low value to $1 - \mathbf{\Sigma}_l^i$ for positive values of $\nabla^i_{\mathbf{h}_l} F^-(\mathbf{x})$. In cases where positive and negative values are not consistent (*i.e.* $\nabla^i_{\mathbf{h}_l} F^+(\mathbf{x})$ and $\nabla^i_{\mathbf{h}_l} F^-(\mathbf{x})$ do not have the same sign), we simply set $\mathbf{\Sigma}_l^i$ to 0.5.

Appendix C. Additional Qualitative Results

In the following, we show additional qualitative results comparing DMBP with previous gradient-based approaches. For each example, we show the attribution maps (third row) and the two linear mappings producing the positive and negative pixel attributions when they are multiplied by the image (first and second rows). For DMBP, the latter are explicitly computed during optimization. For the rest of methods, they are computed by decomposing the obtained linear mapping according to the sign of the obtained pixel-level attributions. The visualization of these linear mappings allow to understand the low-level information (*i.e.* colors, edges, textures) that produces the attributions for each pixel. From the reported results, we can extract the same conclusions than the ones discussed in Section 4.3.

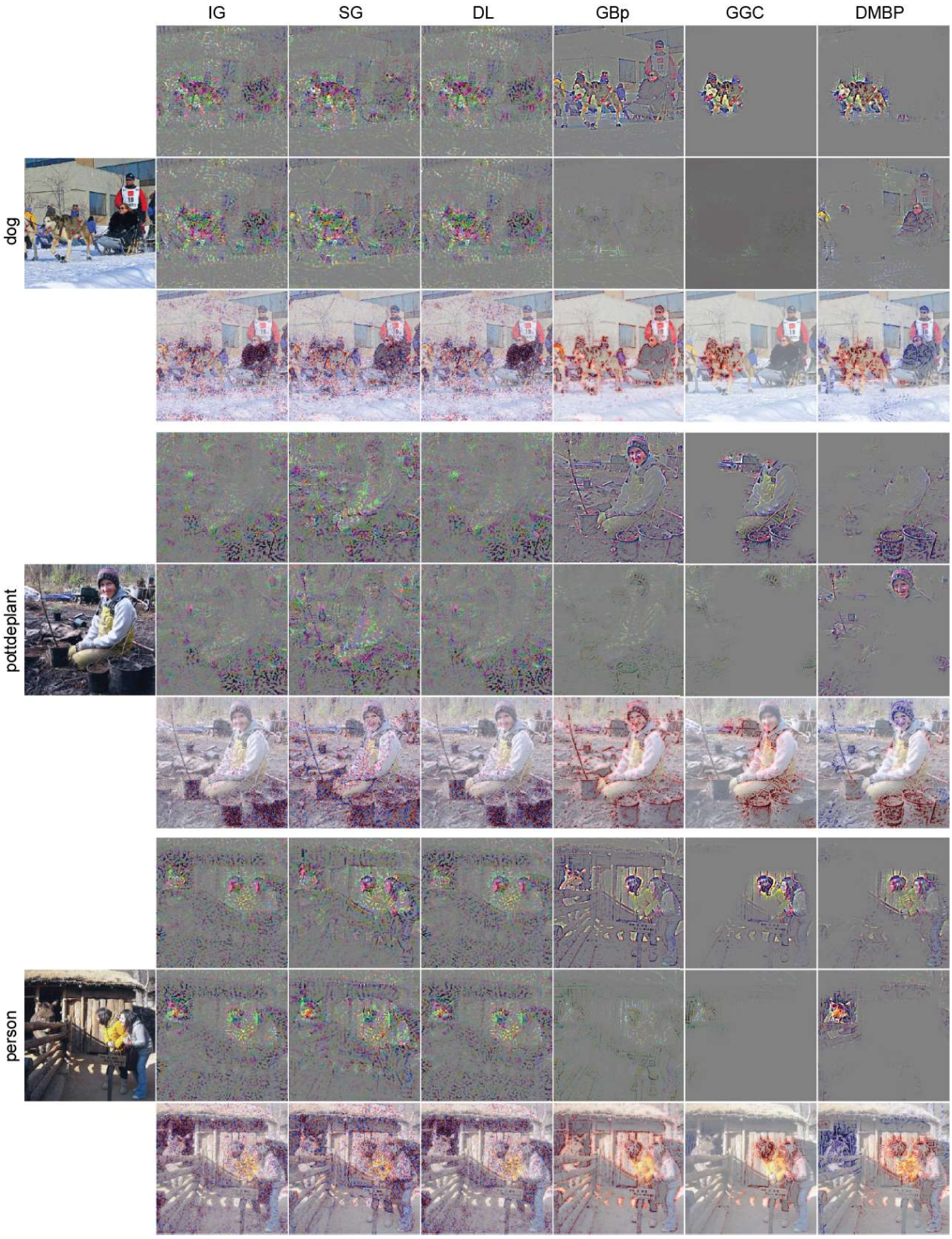

Figure 6. **Qualitative results for DMBP and alternative gradient-based approaches**. VGG16 applied over VOC images.

Figure 7. **Qualitative results for DMBP and alternative gradient-based approaches**. VGG16 applied over VOC images.

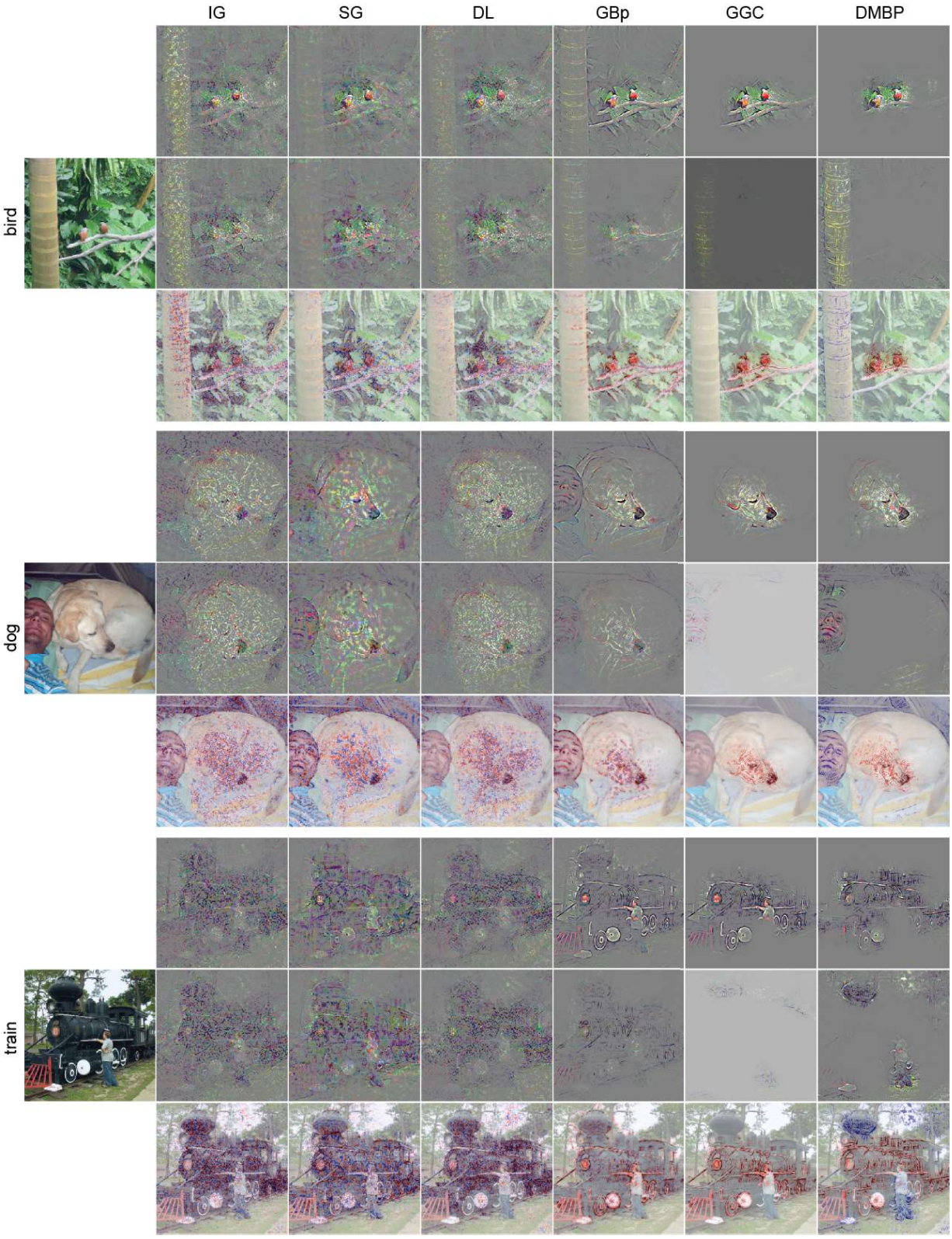

Figure 8. **Qualitative results for DMBP and alternative gradient-based approaches**. ResNet50 applied over VOC images.

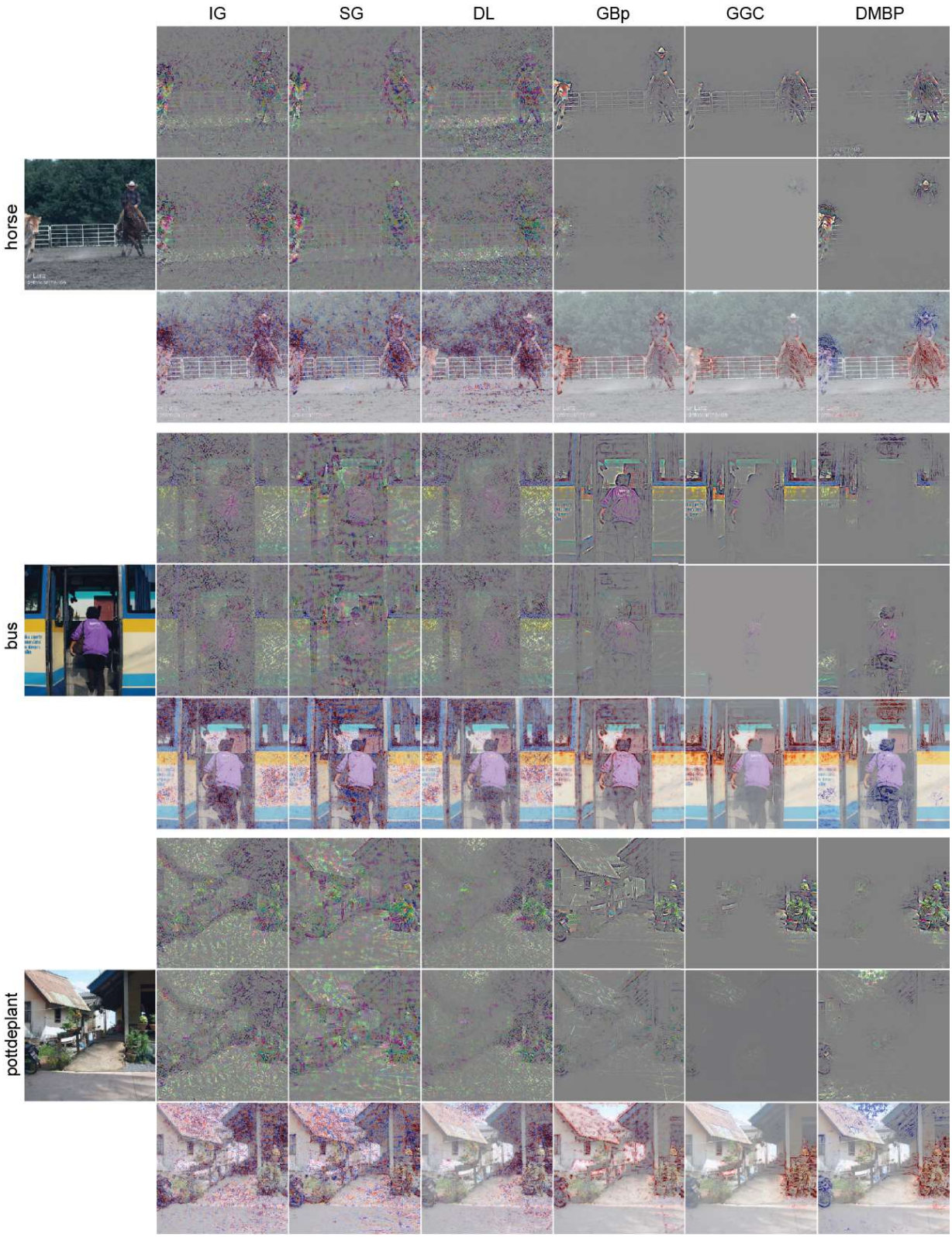

Figure 9. **Qualitative results for DMBP and alternative gradient-based approaches**. ResNet50 applied over VOC images.

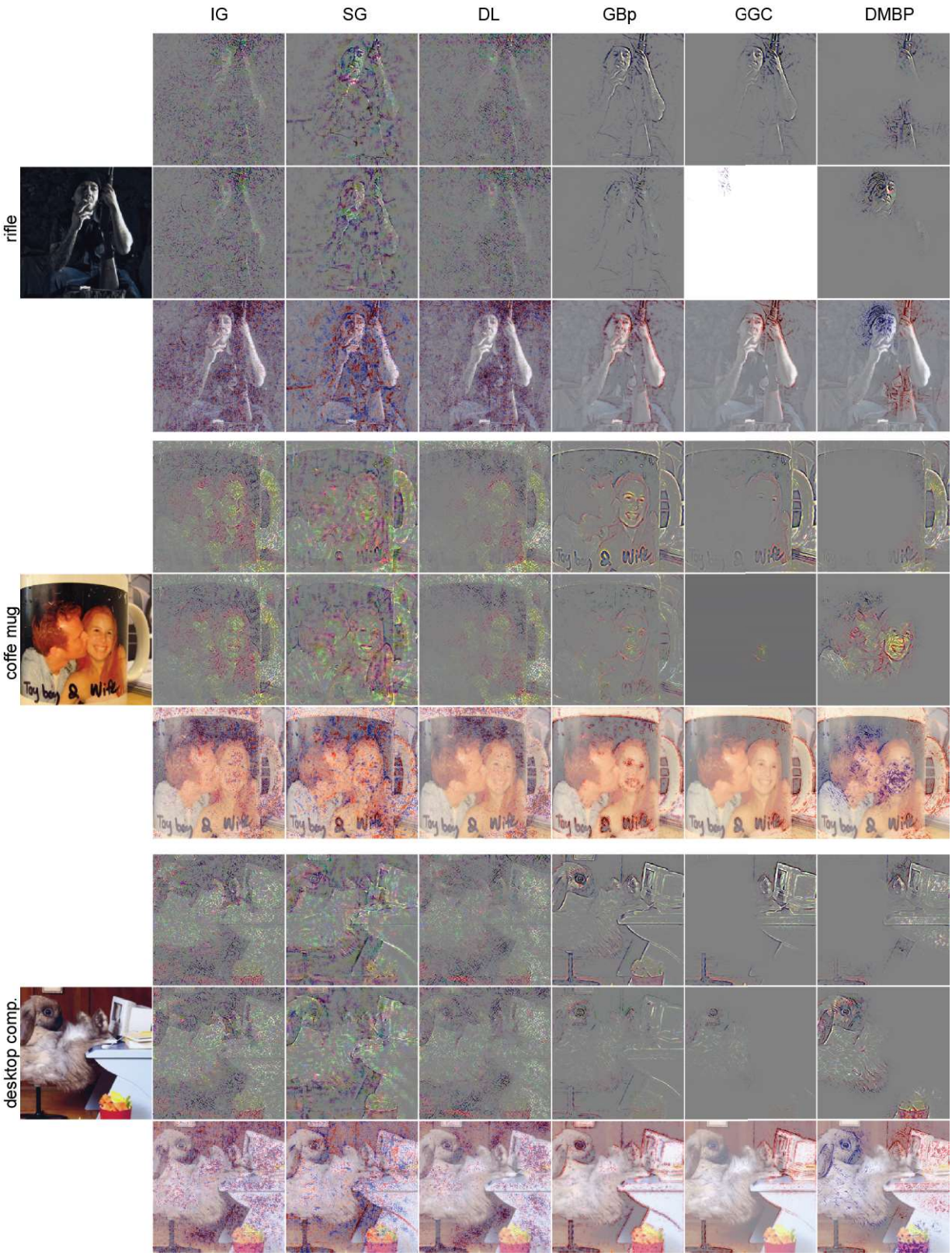

Figure 10. **Qualitative results for DMBP and alternative gradient-based approaches**. ResNet50 applied over ImageNet images.

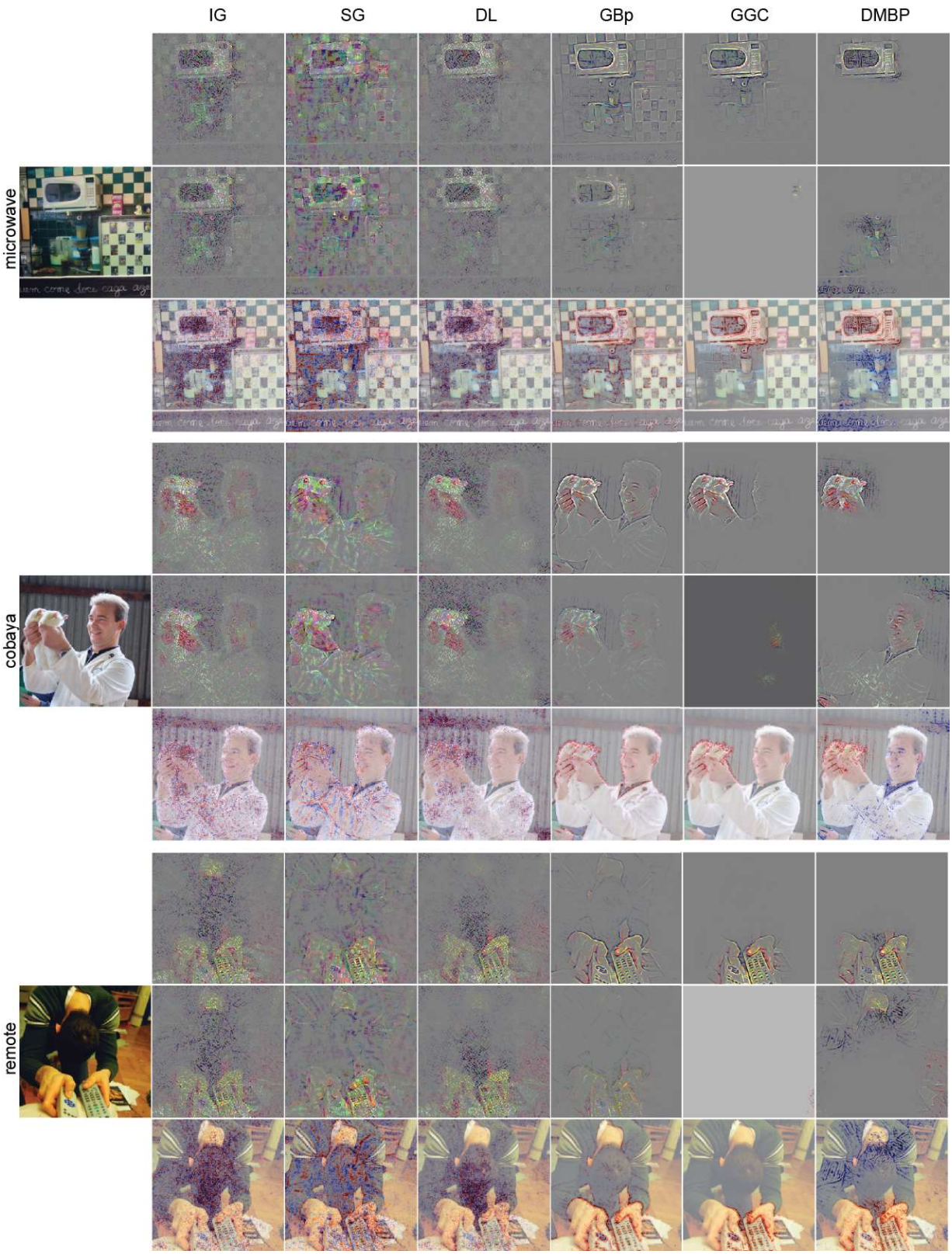

Figure 11. **Qualitative results for DMBP and alternative gradient-based approaches**. ResNet50 applied over ImageNet images.

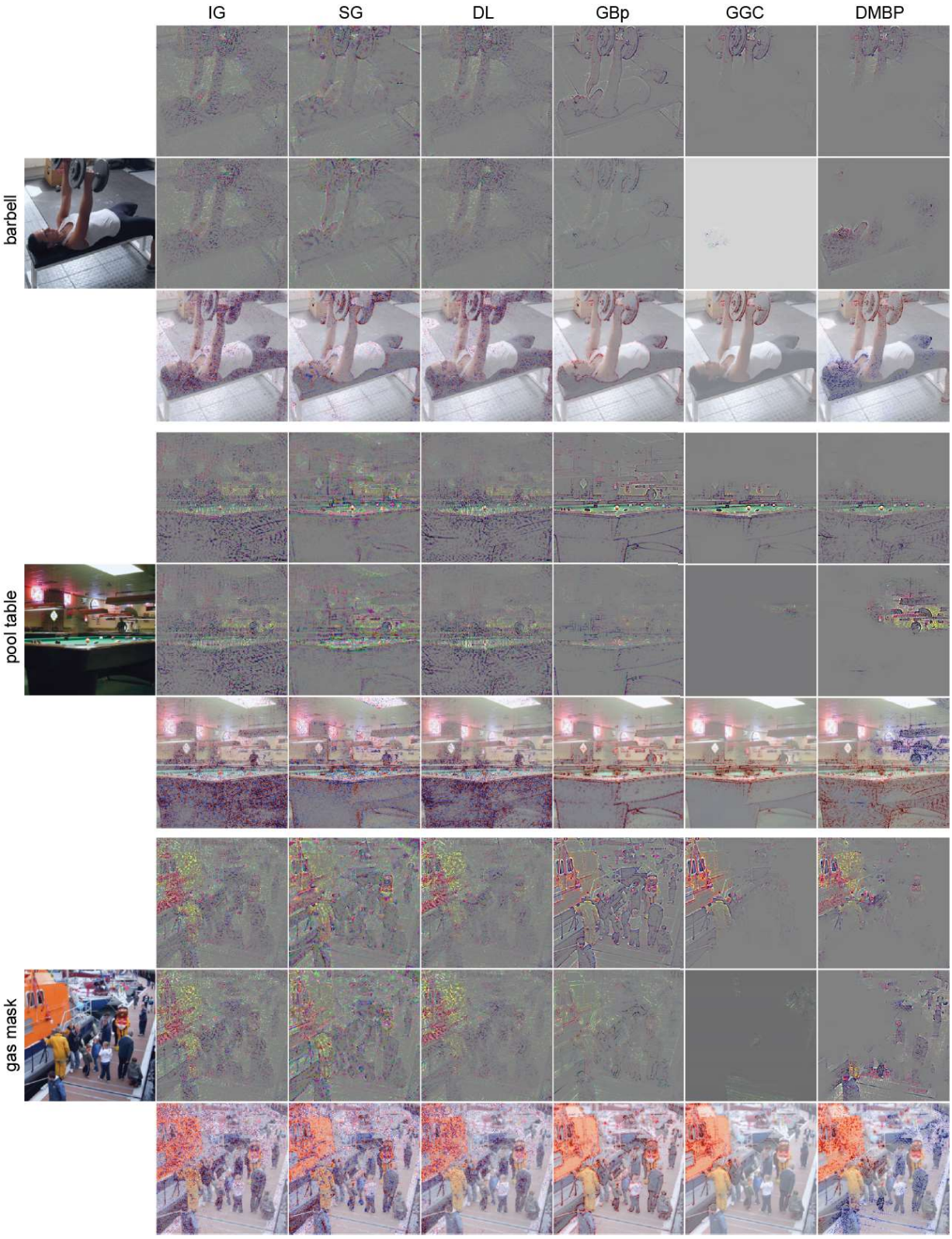

Figure 12. **Qualitative results for DMBP and alternative gradient-based approaches**. VGG16 applied over ImageNet images.

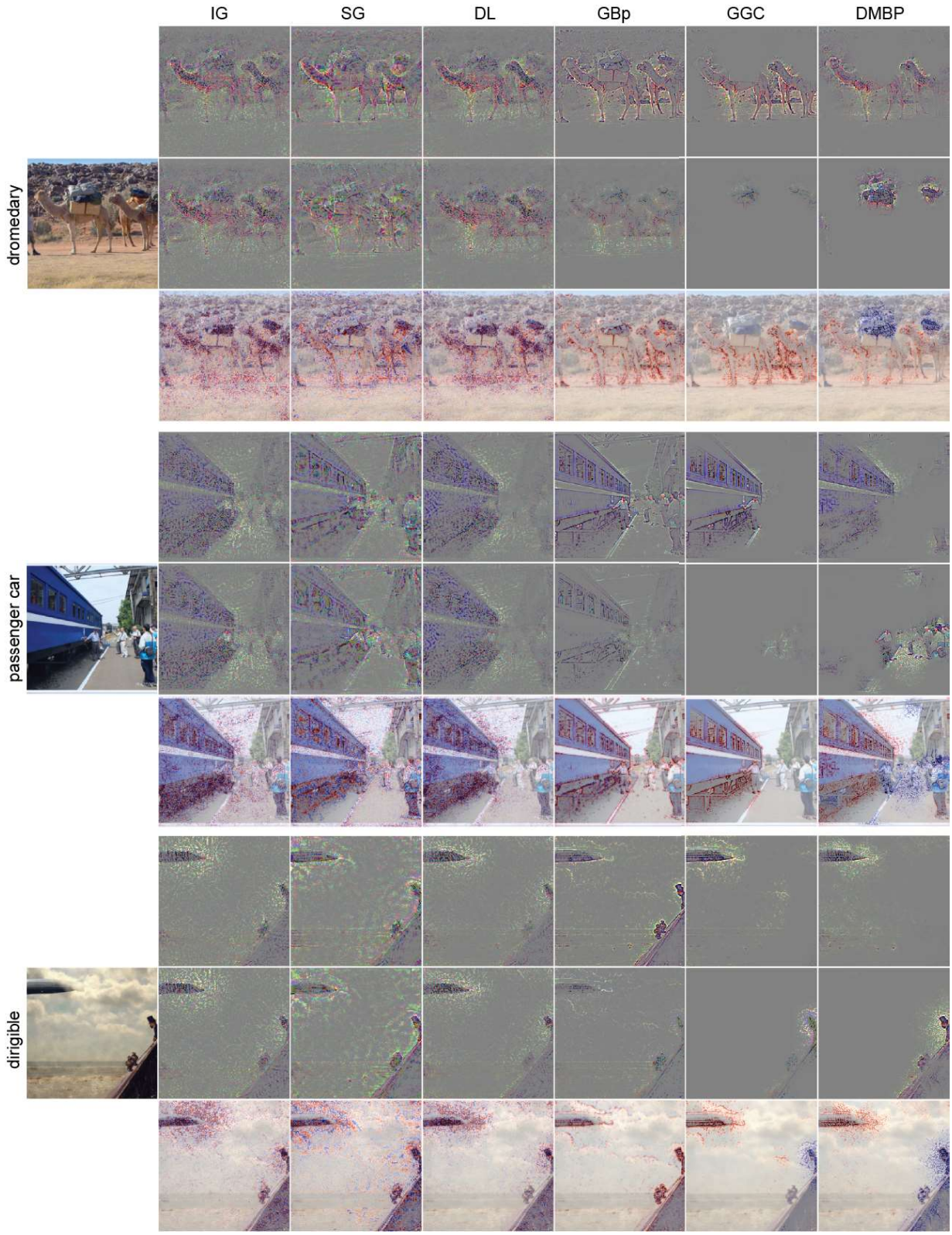

Figure 13. **Qualitative results for DMBP and alternative gradient-based approaches**. VGG16 applied over ImageNet images.