Studying How to Efficiently and Effectively Guide Models with Explanations

Appendix

Table of Contents

In this supplement to our work on using explanations to guide models, we provide:

In this section, we present additional *qualitative* results. In particular, we provide:

In this section, we present additional *quantitative* results. In particular, we show:

In this section, we provide additional results for the Waterbirds-100 dataset. In particular, we provide full results regarding **classification performance** with and without model guidance as well as **additional qualitative visualizations** of the attribution maps.

In this section, we provide relevant implementation details; note that all code will be made available upon publication. In particular, we provide:

Given the large amount of experimental results, in each of the preceding sections we show only a sub-selection of those results for improved readability. In section X, we provide the *full* results across datasets, models, layers, experiments, and metrics, to peruse at the reader's convenience.

A. Additional Qualitative Results (VOC and COCO)

A.1. Qualitative Examples Across Losses, Attribution Methods, and Layers

In Figs. A1 and A2, we visualize attributions across losses, attribution methods, and layers for the same set of examples from the VOC and COCO datasets respectively. As discussed in the main paper, we make the following observations.

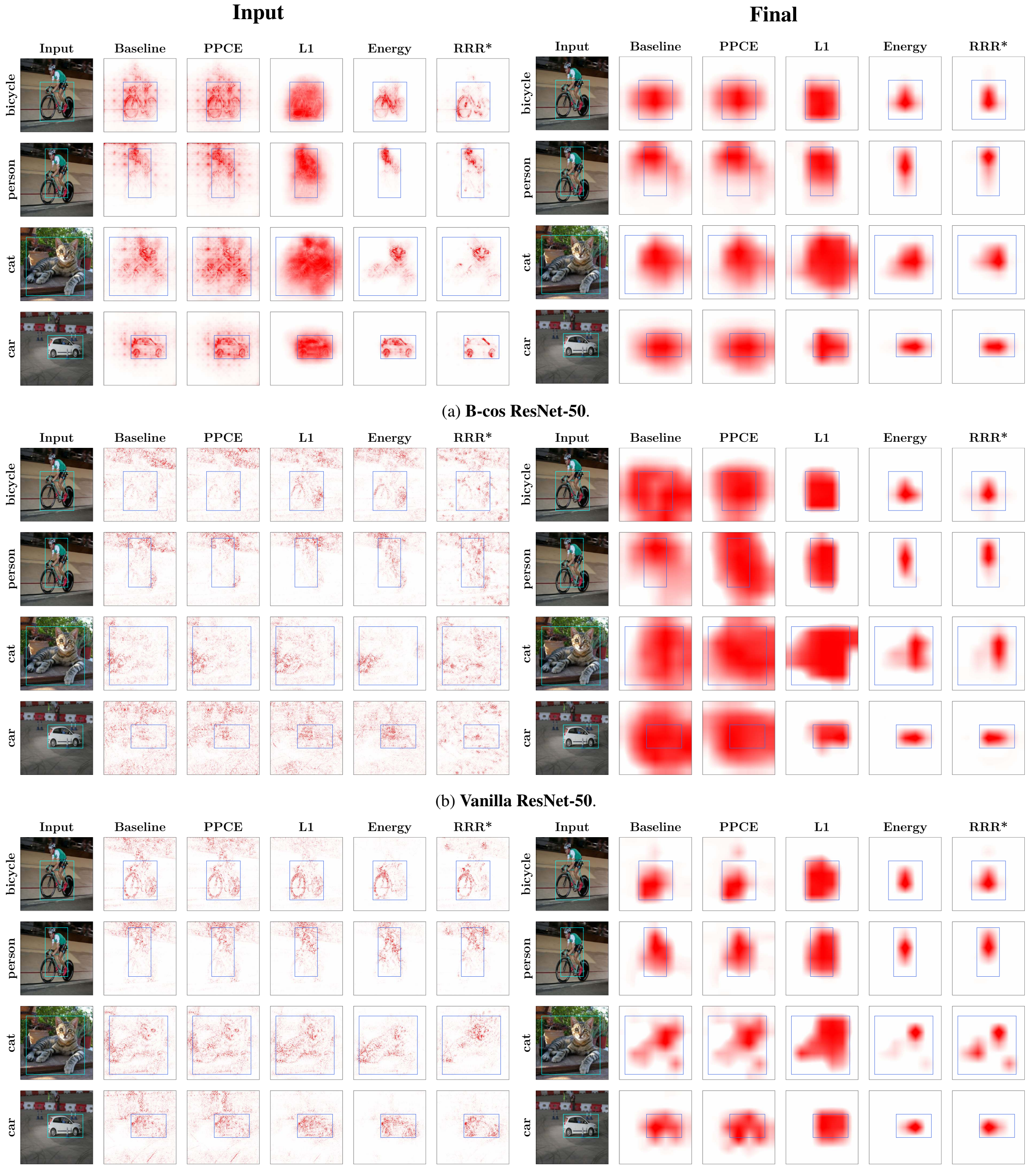
First, when guiding models at the *final layer*, we observe a marked improvement in the granularity of the attribution maps for all losses (**R5**), except for PPCE, for which we do not observe notable differences. The improvements are particularly noticeable on the COCO dataset (Fig. A2, “Final” column), in which the objects tend to be smaller. E.g., when looking at the airplane image (last row per model), we observe much fewer attributions in the background after applying model guidance.

Second, as the L_1 loss optimizes for uniform coverage *within* the bounding boxes, it provides coarser attributions that tend to fill the entire bounding box (cf. **R3**). This can be observed particularly well for the large objects from the VOC dataset: e.g., whereas models trained with the Energy and the RRR loss highlight just a relatively small area within the bounding box of the cat (Fig. A1, ”Final” column, third row), the L_1 loss yields much more distributed attributions for all models.

Third, at the input layer, the B-cos models show the most notable qualitative improvements (cf. **R4**). In particular, although the $\mathcal{X}$-DNN models show some reduction in noisy background attributions (e.g. last rows in Fig. A1c and Fig. A2c), the attributions remain rather noisy for many of the images; for the Vanilla models, the improvements are even less pronounced (Fig. A1b, Fig. A2b). The B-cos models, on the other hand, seem to lend themselves better to such guidance being applied to the attributions at the input layer (Fig. A1a, Fig. A2a) and the resulting attributions show much more detail (Energy + RRR) or an increased focus on the entire bounding box (L_1). Especially with the Energy, the B-cos models are able to clearly focus on even small objects, see Fig. A2a.

For additional results from both the VOC as well as the COCO dataset, please see Fig. A3.

PASCAL VOC 2007.



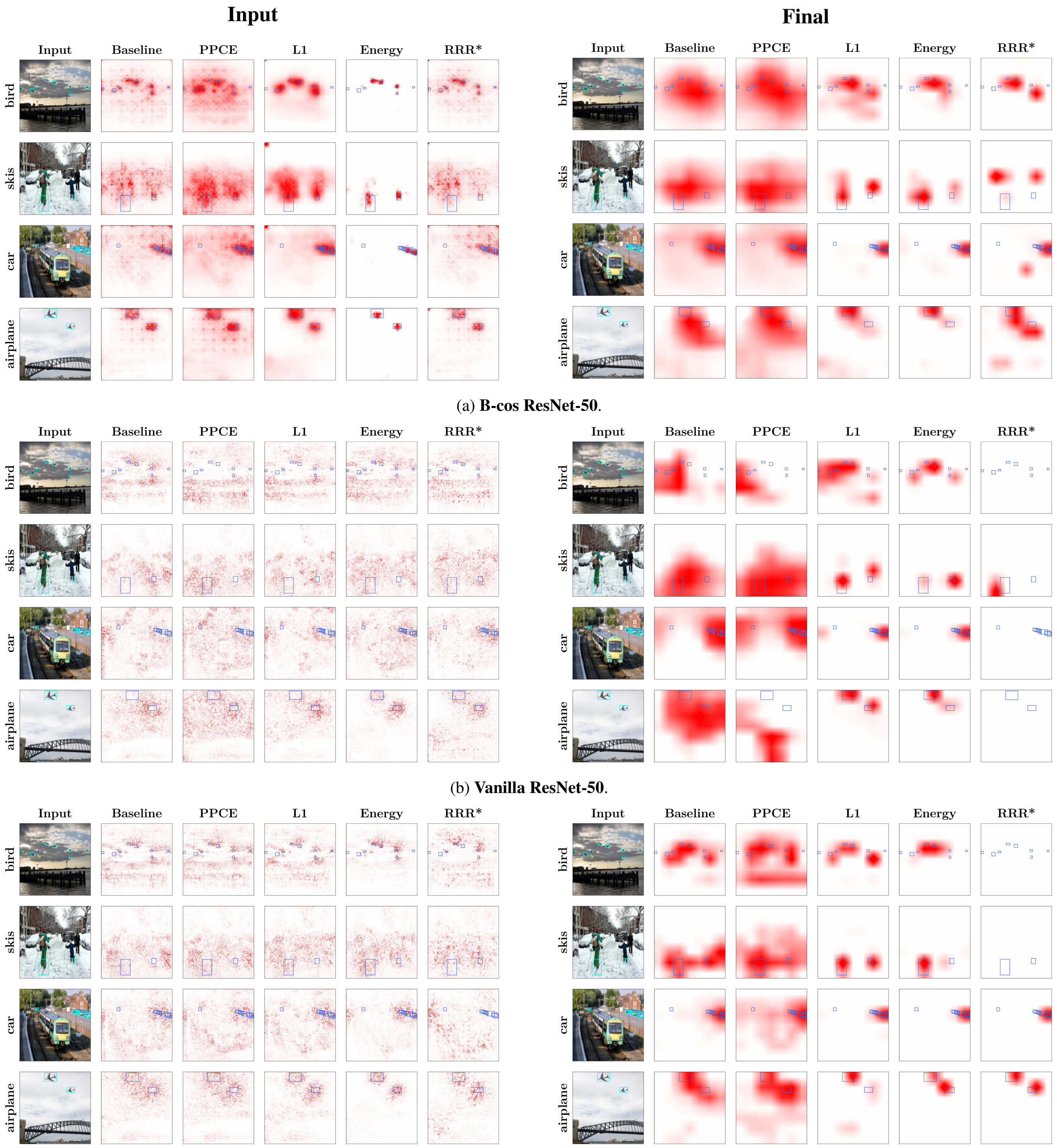
(a) **B-cos ResNet-50**.

(b) **Vanilla ResNet-50**.

(c) $\mathcal{X}$-**DNN ResNet-50**.

Fig. A1: Qualitative examples from the **VOC dataset**. In particular, this figure allows to compare between models (**major rows**, i.e. (a), (b), and (c)) losses (**major columns**) and layers (**left+right**) for multiple images (**minor rows**).

MS COCO 2014.



(a) **B-cos ResNet-50**.

(b) **Vanilla ResNet-50**.

(c) $\mathcal{X}$**-DNN ResNet-50**.

Fig. A2: Qualitative examples from the **COCO dataset**. In particular, this figure allows to compare between models (**major rows**, i.e. (a), (b), and (c)) losses (**major columns**) and layers (**left+right**) for multiple images (**minor rows**).

Additional qualitative examples.

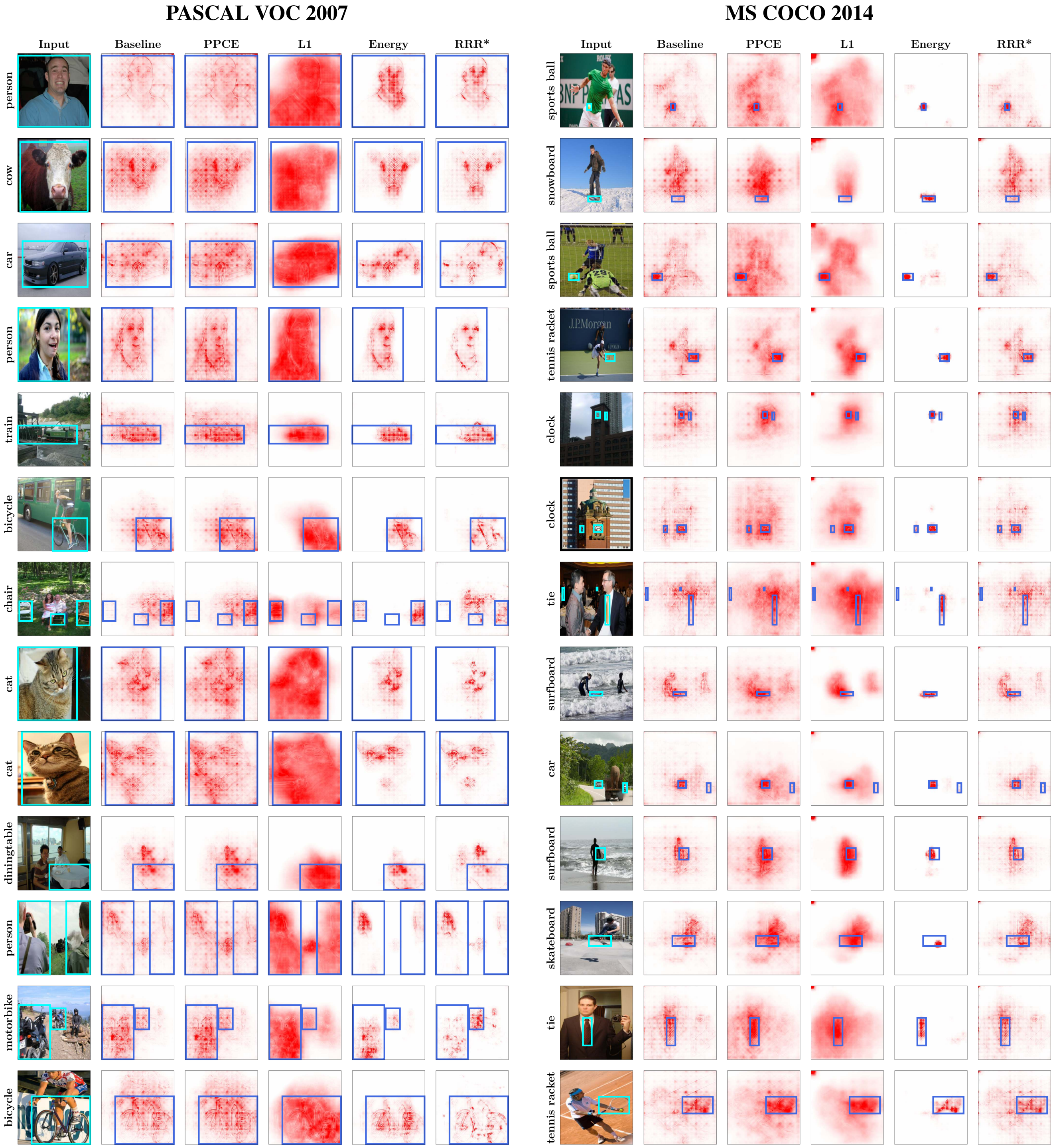


Fig. A3: Qualitative examples from the **VOC (left) and COCO (right)** datasets. In particular, here we just show additional examples for the B-cos models with input attributions, as this configuration exhibits the most detail. We show results for such models trained with different losses (**columns**) for multiple images (**rows**).

A.2. Additional visualizations for training with coarse bounding boxes

In this section, we show more detailed and additional examples of models trained with coarser bounding boxes, i.e. with bounding boxes that are purposefully dilated during training by various amounts (10%, 25%, or 50%), see Fig. A4. In accordance with our findings in the main paper (cf. R8), we observe that the Energy loss is highly robust to such 'annotation errors': the attribution maps improve noticeably in all cases (compare the Energy row with the respective baseline result). In contrast, the L_1 loss seems more dependent on high-quality annotations, which we also observe quantitatively, see Fig. B8.

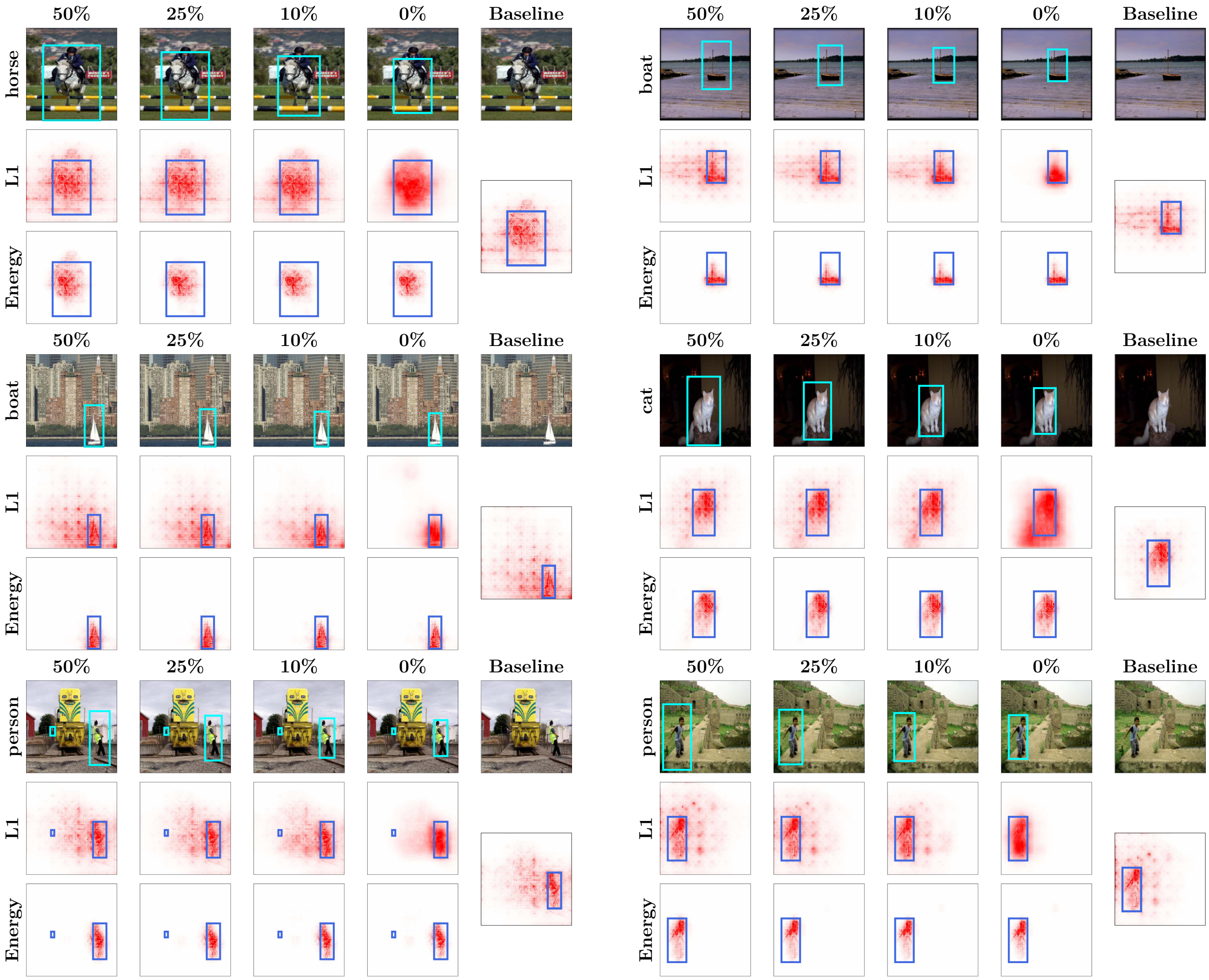


Fig. A4: **Qualitative examples of the impact of using coarse bounding boxes for guidance.** We show examples of B-cos attributions from the input layer on the baseline model and on models guided with the Energy and L_1 localization losses with varying degrees of dilation $\{10\%, 25\%, 50\%\}$ in bounding boxes during training. For each example (**block** in the figure), we show the image and bounding boxes with varying degrees of dilation (**top** row), attributions with the L_1 localization loss (**middle** row), and attributions with the Energy localization loss (**bottom** row). We find that in contrast to using the L_1 localization loss, guidance with Energy localization loss maintains localization of attributions to on-object features even with dilated bounding boxes. Note that bounding boxes are dilated only during training, not during evaluation. Bounding boxes in **light blue** show the extent of dilation that *would have been used* had the image been from the training set, while those in **dark blue** show undilated bounding boxes that are used during evaluation.

B. Additional Quantitative Results (VOC and COCO)

In this section, we provide additional quantitative results from our experiments on the VOC and COCO datasets. Specifically, in Sec. B.1, we show additional results comparing classification and localization performance. In Sec. B.2 we present results for guiding models via GradCAM attributions. In Sec. B.3, we show that training at intermediate layers can be a cost-effective way approach to performing model guidance. In Sec. B.4, we evaluate how well the attributions localize to on-object features (as opposed to background features) within the bounding boxes, and find that the Energy outperforms other localization losses in this regard. In Sec. B.5, we provide additional analyses regarding training with a limited number of annotated images. Finally, in Sec. B.6, we provide additional analyses regarding the usage of coarse, dilated bounding boxes during training.

B.1. Comparing Classification and Localization Performance

In this section, we discuss additional quantitative findings with respect to localization and classification performance metrics (IoU, mAP) for a selected subset of the experiments; for a full comparison of all layers and metrics, please see Figs. X1, X2, X3 and X4.

Additional IoU results. In Figs. B1 and B2, we show the remaining results comparing IoU vs. F1 scores that were not shown in the main paper for VOC and COCO respectively. Similar to the results in the main paper for the EPG metric (Fig. 5), we find that the results between datasets are highly consistent for the IoU metric.

In particular, as discussed in Sec. 5.1, we find that the L_1 loss yields the largest improvements in IoU when optimized at the final layer, see bottom rows of Figs. B1 and B2. At the input layer, we find that Vanilla and $\mathcal{X}$-DNN ResNet-50 models are not improving their IoU scores noticeably, whereas the B-cos models show significant improvements. We attribute this to the noisy patterns in the attribution maps of Vanilla and $\mathcal{X}$-DNN models, which might be difficult to optimize.

IoU results on VOC.

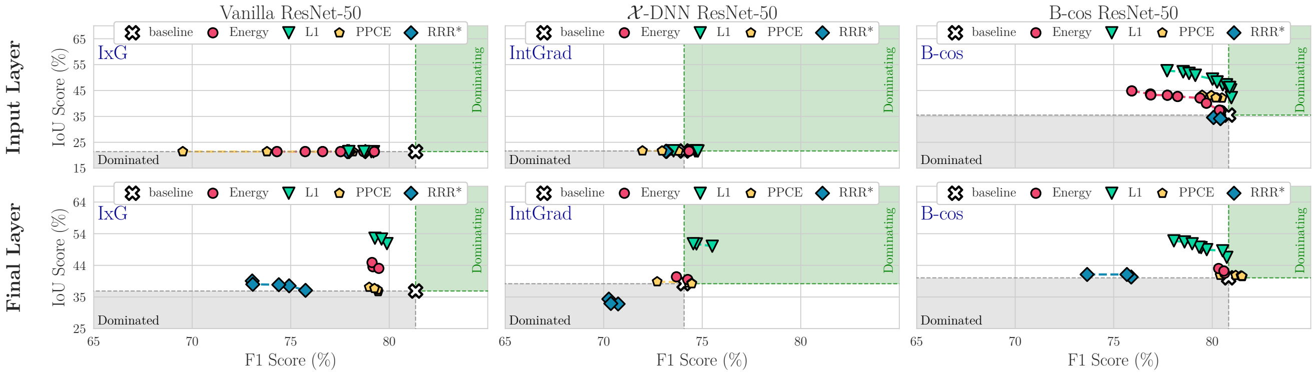


Fig. B1: **IoU results on PASCAL VOC 2007.** We show IoU vs. F1 for all localization loss functions, attribution methods, and layers. In contrast to the consistent improvements observed at the final layer with the L_1 loss, the IoU metric only noticeably improves for the B-cos models after model guidance. We attribute this to the high amount of noise present in the attribution maps of Vanilla and $\mathcal{X}$-DNN models, see e.g. Figs. A1 and A2. For results on the COCO dataset, please see Fig. B2.

Using mAP to evaluate classification performance. In all results so far, we plotted the localization metrics (EPG, IoU) versus the F1 score as a measure of classification performance. In order to highlight that the observed trends are independent of this particular choice of metric, in Fig. B3, we show both EPG as well as IoU results plotted against the mAP score.

In general, we find the results obtained for the mAP metric to be highly consistent with the previously shown results for the F1 metric. E.g., across all configurations, we find the Energy to yield the highest gains in EPG score, whereas the L_1 loss provides the best trade-offs with respect to the IoU metric. In order to easily compare between all results for all datasets and metrics, please see Figs. X1, X2, X3 and X4.

IoU results on COCO.

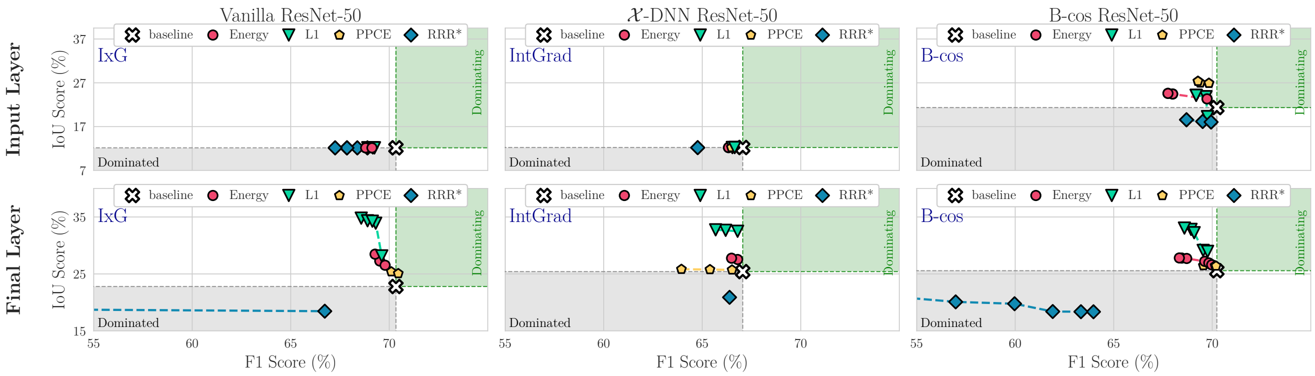
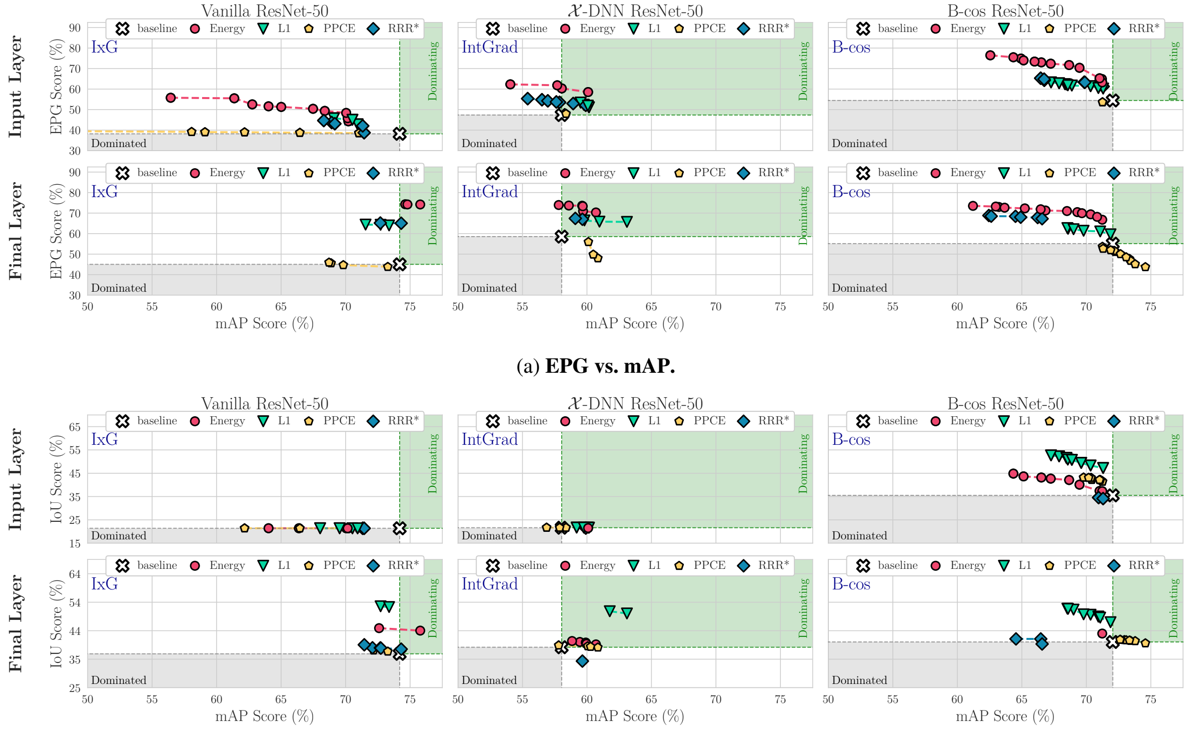


Fig. B2: **IoU results on MS COCO 2014.** We show IoU vs. F1 for all localization loss functions, attribution methods, and layers. In contrast to the consistent improvements observed at the final layer with the L_1 loss, the IoU metric only noticeably improves for the B-cos models after model guidance. We attribute this to the high amount of noise present in the attribution maps of Vanilla and $\mathcal{X}$-DNN models, see e.g. Figs. A1 and A2. For results on the VOC dataset, please see Fig. B1.

Mean Average Precision (mAP) results on VOC.



(a) **EPG vs. mAP.**

(b) **IoU vs. mAP.**

Fig. B3: **Quantitative comparison of EPG and IoU vs. mAP scores for VOC.** To ensure that the trends observed and described in the main paper generalize beyond the F1 metric, in this figure we show the EPG and IoU scores plotted against the mAP metric. In general, we find the results obtained for the mAP metric to be highly consistent with the previously shown results for the F1 metric, see e.g. Figs. 5 and 6. E.g., across all configurations, we find the Energy to yield the highest gains in EPG score, whereas the L_1 loss provides the best trade-offs with respect to the IoU metric. To compare between all results for all datasets and metrics, please see Figs. B3, X1, X2 and X4.

Comparison to GradCAM on VOC.

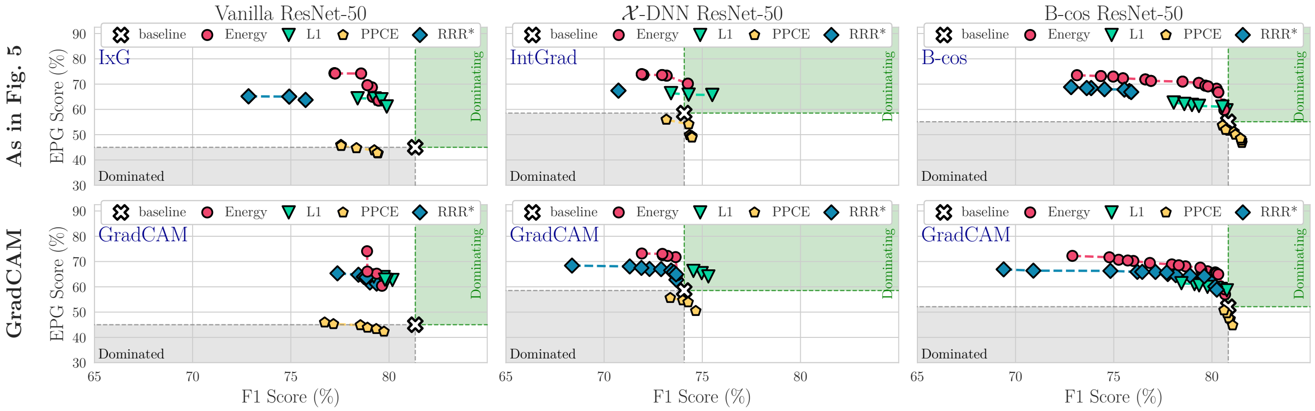


Fig. B4: **Quantitative results using GradCAM.** We show EPG scores vs. F1 scores for all localization losses and models using GradCAM at the final layer (**bottom row**) and compare it to the results shown in the main paper (**top row**). As expected, GradCAM performs very similarly to IxG (Vanilla) and IntGrad ($\mathcal{X}$-DNN) used at the final layer—in particular, note that for ResNet-50 architectures, IxG and IntGrad are very similar to GradCAM for Vanilla and $\mathcal{X}$-DNN models respectively (see Sec. B.2). Similarly, we find GradCAM to also perform comparably to the B-cos explanations when used at the final layer; for IoU results and results on COCO, see Figs. X5 and X6.

B.2. Model Guidance via GradCAM

In Fig. B4, we show the EPG vs. F1 results of training models with GradCAM applied at the final layer on the VOC dataset; for IoU results and results on COCO, please see Figs. X5 and X6. When comparing between rows (**top:** main paper results; **bottom:** GradCAM), it becomes clear that GradCAM performs very similarly to IxG / IntGrad / B-cos attributions on Vanilla / $\mathcal{X}$-DNN / B-cos models. In fact, note that GradCAM is very similar to IxG and IntGrad (equivalent up to an additional zero-clamping) for the respective models and any differences in the results can be attributed to the non-deterministic training pipeline and the similarity between the results should thus be expected.

B.3. Model Guidance at Intermediate Layers

In Sec. 5, we show results for guidance on two 'model depths', i.e. at the input and the final layer. This corresponds to the two depths at which attributions are typically computed, e.g. IxG and IntGrad are typically computed at the input, while GradCAM is typically computed using final spatial layer activations. Following [S12], for a fair comparison we optimize using each attribution methods at identical depths. For the final and intermediate layers in the network, this is done by treating the output activations at that layer as effective inputs over which attributions are to be computed. As done with GradCAM [S15], we then upscale the attribution maps to image dimensions using bilinear interpolation and then use them for model guidance.

In Fig. B5, we show results for performing model guidance at additional intermediate layers: Mid1, Mid2, and Mid3. Specifically, for the ResNet-50 models we use, these layers correspond to the outputs of `conv2_x`, `conv3_x`, and `conv4_x` respectively in the ResNet nomenclature ([S4]), while the final layer corresponds to the output of `conv5_x`. We find that the EPG performance at these intermediate layers through the network follows the trends when moving from the input to the final layer. Similar results for IoU can be found in Fig. X8.

B.4. Evaluating On-Object Localization

The standard EPG metric (Eq. (2)) evaluates the extent to which attributions localize to the bounding boxes. However, since such boxes often include background regions, the EPG score does not distinguish between attributions that focus on the object and attributions that focus on such background regions within the bounding boxes.

To additionally evaluate for on-object localization, we use a variant of EPG that we call On-object EPG. In contrast to standard EPG, we compute the fraction of positive attributions in pixels contained within the segmentation mask of the object out of positive attributions within the bounding box. This measures how well attributions *within the bounding boxes* localize to the object, and is not influenced by attributions outside the bounding boxes. A visual comparison of the two metrics is

EPG results for **intermediate layers** on VOC.

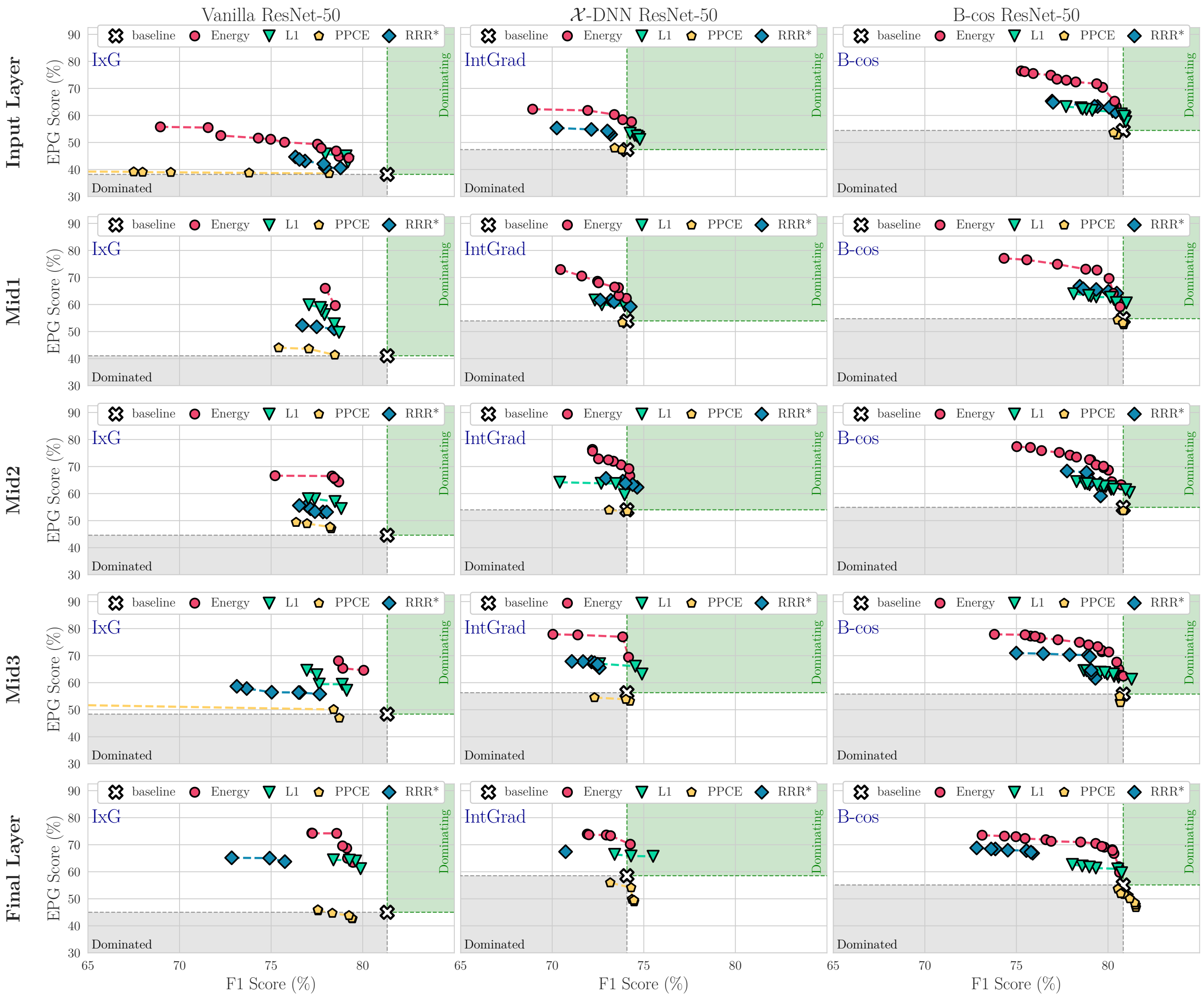


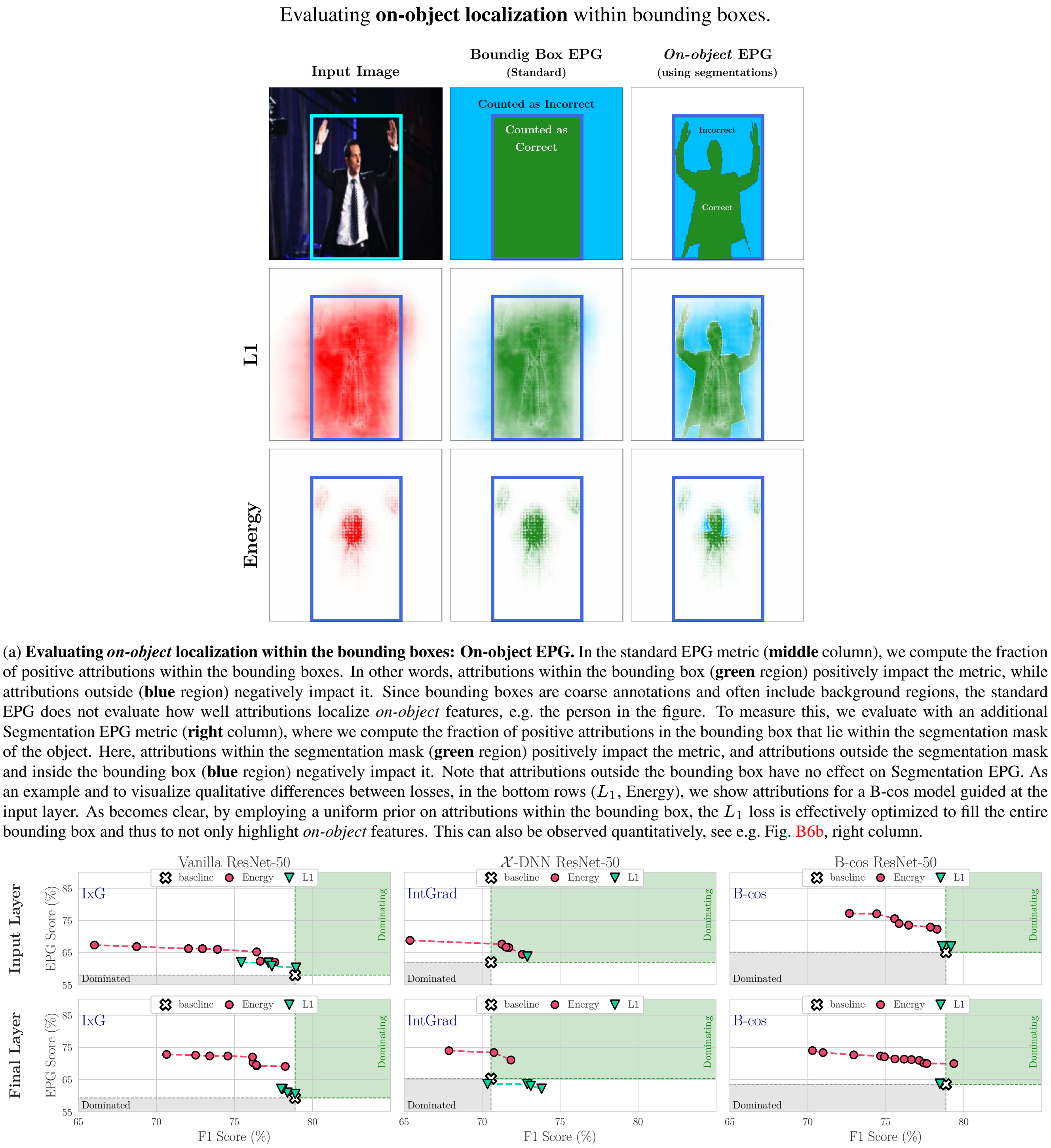
Fig. B5: **Intermediate layer results comparing EPG vs. F1.** We compare the effectiveness of model guidance at varying network depths (**rows**) for each attribution method and model (**columns**) across localization loss functions. For the B-cos model, we find similar trends at all network depths, with the Energy localization loss outperforming all other losses. For the Vanilla and $\mathcal{X}$-DNN models, the Energy loss similarly performs the best, but we also observe improved performance across losses when optimizing at deeper layers of the network. Full results can be found in Figs. X7 and X8.

shown in Fig. B6.

We find that the Energy localization loss outperforms the L_1 localization loss both qualitatively (Fig. B6a) and quantitatively (Fig. B6b) on this metric. This is explained by the fact that the L_1 promotes uniformity in attributions across the bounding box, giving equal importance to on-object and background features within the box. In contrast, the Energy loss only optimizes for attributions to lie within the box, without any constraint on *where* in the box they lie. This also corroborates our previous qualitative observations (e.g. Fig. 9).

B.5. Model Guidance with Limited Annotations

In Fig. B7, we show the impact of using limited annotations when training (Sec. 5.4) when optimizing with the Energy and L_1 localization losses for B-cos attributions at the input. We find that in addition to EPG, trends in IoU scores also remain consistent even when using bounding boxes for just 1% of the the training images.



(a) **Evaluating *on-object* localization within the bounding boxes: On-object EPG.** In the standard EPG metric (**middle** column), we compute the fraction of positive attributions within the bounding boxes. In other words, attributions within the bounding box (**green** region) positively impact the metric, while attributions outside (**blue** region) negatively impact it. Since bounding boxes are coarse annotations and often include background regions, the standard EPG does not evaluate how well attributions localize *on-object* features, e.g. the person in the figure. To measure this, we evaluate with an additional Segmentation EPG metric (**right** column), where we compute the fraction of positive attributions in the bounding box that lie within the segmentation mask of the object. Here, attributions within the segmentation mask (**green** region) positively impact the metric, and attributions outside the segmentation mask and inside the bounding box (**blue** region) negatively impact it. Note that attributions outside the bounding box have no effect on Segmentation EPG. As an example and to visualize qualitative differences between losses, in the bottom rows (L_1, Energy), we show attributions for a B-cos model guided at the input layer. As becomes clear, by employing a uniform prior on attributions within the bounding box, the L_1 loss is effectively optimized to fill the entire bounding box and thus to not only highlight *on-object* features. This can also be observed quantitatively, see e.g. Fig. B6b, right column.

(b) **On-object EPG results.** We evaluate across models (**columns**) and layers (**rows**) for the Energy and L_1 localization losses. As seen qualitatively (e.g. Fig. 9), we find that the Energy loss is more effective than the L_1 loss in localizing attributions to the object as opposed to background regions within the bounding boxes. This is explained by the fact that the L_1 loss promotes uniformity in attributions within the bounding box, and treats both on-object and background features inside the box with equal importance, while the Energy loss only optimizes for attributions to lie within the bounding box without placing any constraints on where they may lie, leaving the model free to decide which regions within the box are important for its decision.

Fig. B6: **Evaluating *on-object* localization via EPG.** We show **(a)** the schema for the on-object EPG metric and how it differs the standard bounding box EPG metric, and **(b)** quantitative results on evaluating with on-object EPG.

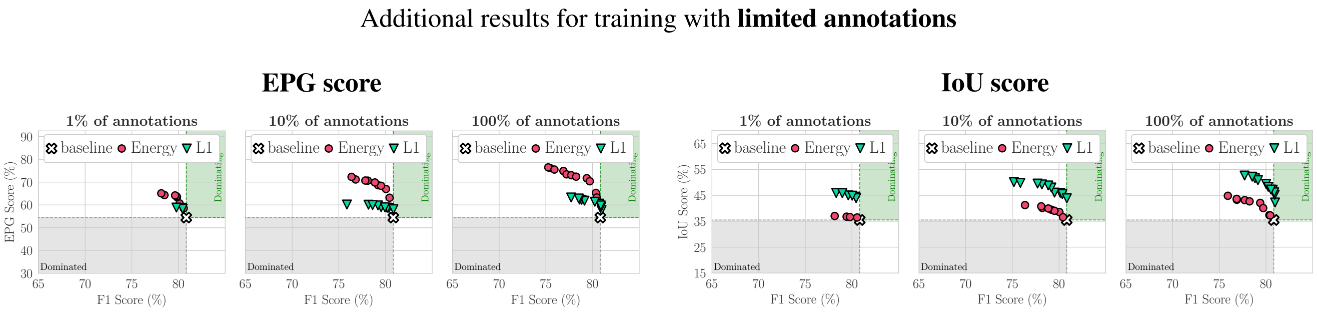
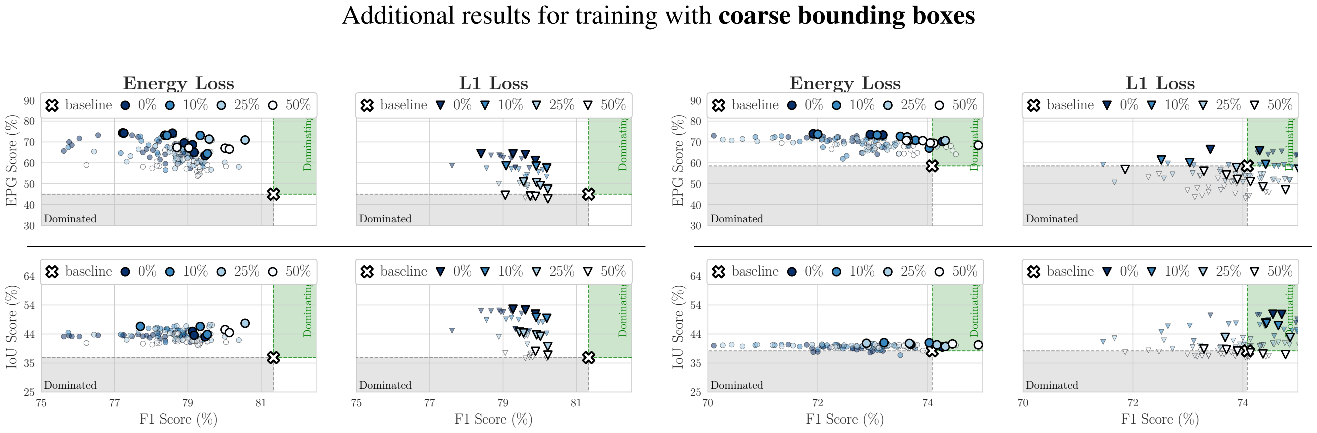


Fig. B7: **EPG and IoU scores for limited annotations.** We show EPG vs. F1 (**left**) and IoU vs. F1 (**right**) for B-cos attributions at the input when optimizing with the Energy and L_1 localization losses, when using $\{1\%, 10\%, 100\%\}$ training annotations. We find that model guidance is generally effective even when training with annotations for a limited number of images. While the performance slightly worsens when using 1% annotations, using just 10% annotated images yields similar gains to using a fully annotated training set. Full results can be found in Figs. X9 and X10.

B.6. Model Guidance with Noisy Annotations



(a) **Vanilla ResNet-50 @ Final.**

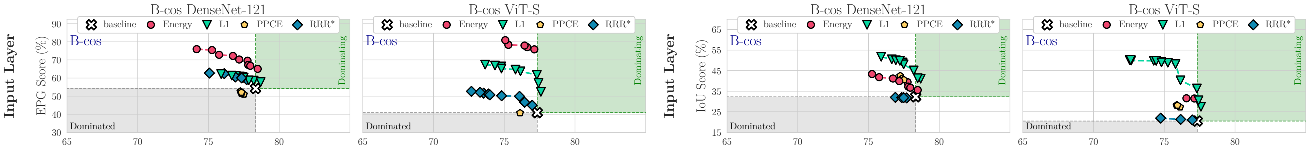
(b) **$\mathcal{X}$-DNN ResNet-50 @ Final.**

Fig. B8: **Coarse bounding box results.** We show the impact of dilating bounding boxes during training for the **(a)** Vanilla and **(b)** $\mathcal{X}$-DNN models. Similar to the results seen with B-cos models (Fig. 10), we find that the Energy localization loss is generally robust to coarse annotations, while the effectiveness of guidance with the L_1 localization loss worsens as the extent of coarseness (dilations) increases. Full results in Fig. X11.

In Fig. B8, we additionally show the impact of training with coarse, dilated bounding boxes for IxG attributions on the Vanilla model, and IntGrad attributions on the $\mathcal{X}$-DNN model. Similar to the results seen with B-cos attributions (Fig. 10), we find that the Energy localization loss is robust to coarse annotations, while the performance with L_1 localization loss worsens as the dilations increase.

B.7. Evaluation on DenseNet and ViT models

In Fig. B9, we evaluate the best performing configurations from our study, i.e. performing guidance using B-cos attributions at input, on additional model backbones, specifically DenseNet-121 and ViT-S. We find that the trends observed with ResNet-50 models generalizes to these backbones, with the Energy loss yielding the highest gains for EPG, and the L_1 loss yielding the highest gains for IoU.



(a)

(b)

Fig. B9: **EPG and IoU vs. F1 on VOC for two additional B-cos architectures.** We find that the trends observed in the main paper for a B-cos ResNet-50 backbone (cf. Figs. 5 and 6, right columns) generalize to other backbone architectures. In particular, we find that the L_1 loss yields the highest gains in IoU, whereas the Energy loss yields the highest gains in EPG, both for a DenseNet-121 and a ViT-S model.

C. Waterbirds Results

			Conventional Setting					Reversed Setting				
	Layer	Loss	G1 Acc	G2 Acc	G3 Acc	G4 Acc	Overall	G1 Acc	G2 Acc	G3 Acc	G4 Acc	Overall
B-cos	Input	Energy	99.2 (±0.1)	40.4 (±1.0)	**56.1 (±4.0)**	96.6 (±0.4)	**71.2 (±0.1)**	**99.4 (±0.1)**	**70.2 (±2.1)**	**62.8 (±2.1)**	96.5 (±0.6)	**83.6 (±1.1)**
		L1	99.3 (±0.1)	37.0 (±0.8)	51.1 (±1.9)	**97.2 (±0.3)**	69.5 (±0.2)	99.3 (±0.3)	67.7 (±3.3)	58.8 (±5.0)	**96.7 (±0.7)**	82.2 (±0.9)
	Final	Energy	99.3 (±0.1)	**41.0 (±2.1)**	53.1 (±0.8)	96.3 (±0.5)	71.1 (±0.9)	**99.4 (±0.2)**	70.1 (±3.1)	60.2 (±3.9)	95.8 (±1.1)	83.2 (±1.1)
		L1	99.3 (±0.1)	34.3 (±3.2)	49.4 (±2.6)	96.6 (±0.6)	68.2 (±1.1)	**99.4 (±0.1)**	69.8 (±2.1)	56.3 (±1.8)	96.1 (±0.7)	82.8 (±0.8)
	Baseline		**99.4 (±0.1)**	37.2 (±0.2)	43.4 (±2.4)	96.5 (±0.1)	68.7 (±0.2)	**99.4 (±0.1)**	62.8 (±0.2)	56.6 (±2.4)	96.5 (±0.1)	80.1 (±0.2)
$\mathcal{X}$-DNN	Input	Energy	99.3 (±0.2)	**47.0 (±9.1)**	49.2 (±4.8)	**96.8 (±0.7)**	**73.1 (±3.4)**	99.0 (±0.3)	**67.6 (±4.8)**	**63.9 (±3.6)**	96.1 (±0.7)	**82.6 (±2.0)**
		L1	99.1 (±0.6)	40.4 (±7.3)	41.8 (±3.8)	96.5 (±0.6)	69.6 (±3.2)	**99.3 (±0.2)**	59.1 (±4.7)	63.6 (±6.1)	96.0 (±0.9)	79.3 (±1.3)
	Final	Energy	99.2 (±0.4)	42.5 (±10.4)	**54.2 (±3.2)**	96.6 (±0.9)	71.9 (±4.2)	99.2 (±0.2)	65.3 (±2.0)	62.3 (±3.3)	96.0 (±0.5)	81.5 (±0.9)
		L1	**99.4 (±0.1)**	45.1 (±4.0)	42.8 (±2.8)	96.5 (±0.5)	71.7 (±1.4)	**99.3 (±0.2)**	62.9 (±4.8)	59.8 (±4.8)	95.8 (±0.7)	80.4 (±1.8)
	Baseline		99.3 (±0.1)	39.8 (±0.7)	38.6 (±2.5)	96.3 (±0.7)	69.1 (±0.6)	**99.3 (±0.1)**	60.2 (±0.7)	61.4 (±2.5)	**96.3 (±0.7)**	79.6 (±0.5)
Vanilla	Input	Energy	99.4 (±0.2)	42.4 (±2.6)	47.9 (±3.5)	97.1 (±0.4)	71.2 (±1.0)	**99.6 (±0.2)**	50.7 (±7.3)	52.4 (±1.7)	97.2 (±0.5)	75.1 (±2.9)
		L1	**99.5 (±0.1)**	46.1 (±4.4)	51.1 (±4.0)	97.5 (±0.1)	73.1 (±1.6)	**99.6 (±0.1)**	48.0 (±7.8)	49.7 (±3.7)	96.8 (±0.6)	73.7 (±2.7)
	Final	Energy	**99.5 (±0.0)**	56.1 (±7.0)	**60.7 (±5.5)**	97.0 (±0.5)	**78.1 (±2.6)**	99.5 (±0.1)	59.4 (±5.9)	**56.5 (±3.7)**	97.2 (±0.5)	**78.9 (±1.9)**
		L1	**99.5 (±0.1)**	**57.1 (±2.9)**	55.4 (±2.5)	96.7 (±0.6)	77.8 (±1.0)	99.5 (±0.1)	56.3 (±6.7)	51.6 (±3.1)	97.3 (±0.6)	77.1 (±2.5)
	Baseline		99.4 (±0.0)	39.6 (±0.7)	53.7 (±2.1)	**97.7 (±0.0)**	70.8 (±0.0)	99.4 (±0.0)	**60.4 (±0.7)**	46.3 (±2.1)	**97.7 (±0.0)**	78.1 (±0.1)

Table C1: **Classification performance on Waterbirds** after model guidance with the L_1 and the Energy loss. We find that both losses consistently improve the models' classification performance over the baseline model (i.e. a model without guidance). These improvements are particularly pronounced in the groups *not seen during training*, i.e. landbirds on water ("G2") and waterbirds on land ("G3"). For qualitative visualizations of the effect of model guidance on the waterbirds dataset, see Fig. C1.

As discussed in section Sec. 5.5, we use the Waterbirds-100 dataset [S48, S11] to evaluate the effectiveness of model guidance in a setting where strong spurious correlations are present in the training data. This dataset consists of four groups—*Landbird* on *Land* (**G1**), *Landbird* on *Water* (**G2**), *Waterbird* on *Land* (**G3**), and *Waterbird* on *Water* (**G4**)—of which only groups **G1** and **G4** appear during training and the background is thus perfectly correlated with the type of bird (e.g. Landbird on land).

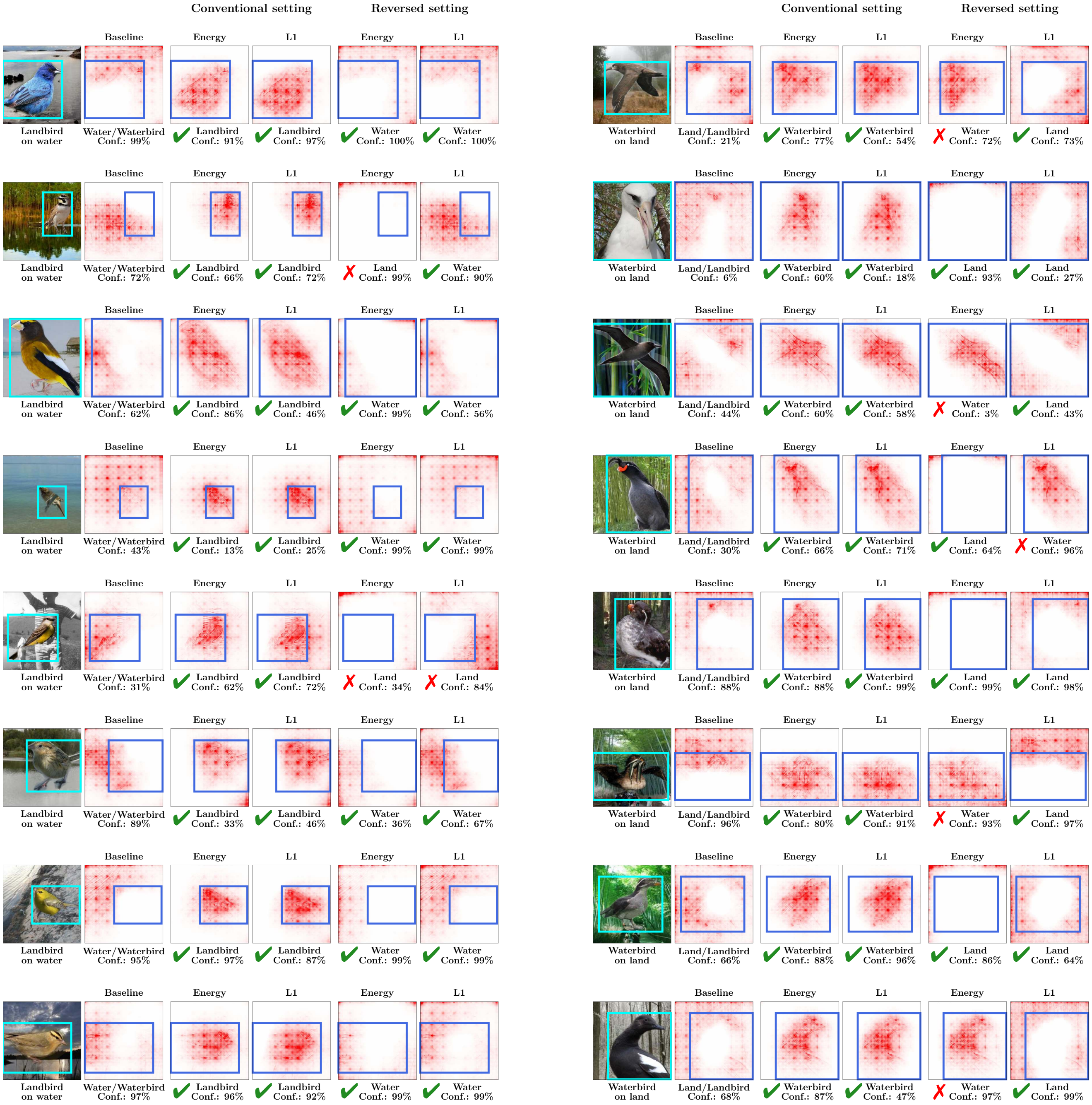
To evaluate the effectiveness of model guidance, we train the models on two binary classification tasks: to classify the type of birds (the *conventional setting*) or the background (the *reversed setting*, as described in [S11]) and evaluate models without guidance (baselines), as well as with guidance: specifically, for guiding the models, we evaluate different models (Vanilla, $\mathcal{X}$-DNN, B-cos) with different guidance losses (Energy, L_1) applied at different layers (Input and Final), see Tab. C1. For each model, we use its corresponding attribution method, i.e. IxG for Vanilla, IntGrad for $\mathcal{X}$-DNN, and B-cos for B-cos.

In Tab. C1 we present the classification performance for the individual groups (**G1-G4**) as well as the average over all samples ('Overall') across all configurations; note that the group sizes differ in the test set and the average over the individual group acccuracies thus differs from the overall accuracy. For each row, the results are averaged over 4 runs (2 random training seeds and 2 different sets of 1% annotated samples) with the exception of the baseline results being an average over 2 runs.

In almost all cases, we find that both of the evaluated losses (Energy, L_1) improve the models' classification performance over the baseline. As expected, these improvements are particularly pronounced in the groups not seen during training, i.e. landbirds on water (**G2**) and waterbirds on land (**G3**).

Further, in Fig. C1, we show attribution maps of the baseline models, as well as the guided models. As can be seen, model guidance not only improves the accuracy, but is also reflected in the attribution maps: e.g., in row 1 of Fig. C1a, we see that while the baseline model originally focused on the background (water) to classify the image, it is possible to guide the model to use the desired features (i.e. the bird in conventional setting and the background in the reversed setting) and consequently arrive at the desired classification decision. As this guidance is 'soft', we also observe cases in which the model still focused on the wrong feature and thus arrived at the wrong prediction: e.g. in Fig. C1b row 1 (reversed setting), the Energy-guided model still focuses on the bird and thus incorrectly predicts 'Water', similar to the L_1-guided model in row 4.

Additional qualitative results on the Waterbirds-100 dataset.



(a) **Landbirds on Water.**

(b) **Waterbirds on Land.**

Fig. C1: **Qualitative results for the Waterbirds dataset.** Specifically, we show input layer attributions for B-cos models trained without guidance ('Baseline') as well as guided via the Energy or L_1 loss. We find that model guidance can be effective both for focusing on the bird and the background. For example, in the top row of (a), the model originally focuses on the background (col. 2) and classifies the image (col. 1) as Water/Waterbird. In the conventional setting, both the Energy and L_1 localization losses are effective in redirecting the focus to the bird (cols. 3-4), changing the model's prediction to Landbird with high confidence. Similarly, in the reversed setting, both localization losses direct the focus to the background (cols. 5-6), which increases the model's confidence in classifying the image as Water.

D. Implementation Details

D.1. Training and Evaluation Details

Implementations: We implement our code using PyTorch[4] [S10]. The PASCAL VOC 2007 [S3] and MS COCO 2014 [S8] datasets and the Vanilla ResNet-50 model were obtained from the Torchvision library[5] [S10, S9]. Official implementations were used for the B-cos[6] [S2] and $\mathcal{X}$-DNN[7] [S5] networks. Some of the utilities for data loading and evaluation were derived from NN-Explainer[8] [S16], and for visualization from the Captum library[9] [S7].

D.1.1 Experiments with VOC and COCO

Training baseline models: We train starting from models pre-trained on ImageNet [S13]. We fine-tune with fixed learning rates in $\{10^{-3}, 10^{-4}, 10^{-5}\}$ using an Adam optimizer [S6] and select the checkpoint with the best validation F1-score. For VOC, we train for 300 epochs, and for COCO, we train for 60 epochs.

Training guided models: We train the models jointly optimized for classification and localization (Eq. (1)) by fine-tuning the baseline models. We use a fixed learning rate of 10^{-4} and a batch size of 64. For each configuration (given by a combination of attribution method, localization loss, and layer), we train using three different values of λ_{loc}, as detailed in Tab. D1. For VOC, we train for 50 epochs, and for COCO, we train for 10 epochs.

Selecting models to visualize: As described in Sec. 4, we select and evaluate on the set of Pareto-dominant models for each configuration after training. Each model on the Pareto front represents the extent of trade-off made between classification (F1) and localization (EPG) performance. In practice, the 'best' model to choose would depend on the requirements of the end user. However, to evaluate the effectiveness of model guidance (e.g. Figs. 1, 2 and 9), we select a representative model on the front whose attributions we visualize. This is done by selecting the model with the highest EPG score with an at most 5 p.p. drop in F1-score.

Efficient Optimization: As described in Sec. 3.5, for each image in a batch, we optimize for localization of a single class selected at random. This approximation allows us to perform model guidance efficiently and keeps the training cost tractable. However, to accurately evaluate the impact of this optimization, we evaluate the localization of all classes in the image at test time.

Training with Limited Annotations: As described in Sec. 5.4, we show that training with a limited number of annotations can be a cost effective way of performing model guidance. In order to maintain the relative magnitude of $\mathcal{L}_{\text{loc}}$ as compared to $\mathcal{L}_{\text{class}}$ in this setting, we scale up the values of λ_{loc} when training. The values of λ_{loc} we use are shown in Tab. D2.

D.1.2 Experiments with Waterbirds-100

Data distributions: The conventional binary classification task includes classifying *Landbird* from *Waterbird*, irrespective of their backgrounds. We use the same splits generated and published by [S11]. As discussed in Sec. C, at training time there are no samples from **G2** or **G3**, making the bird type and the background perfectly correlated. In contrast, both the validation and test sets are balanced across foregrounds and backgrounds, i.e. a landbird is equally likely to occur on land or water, and vice-versa. However, as noted by [S14], using a validation set with the same distribution as the test set leaks information on the test distribution in the process of hyperparameter and checkpoint selection during training. Therefore, we modify the validation split to avoid such information leakage; in particular, we use a validation set with the same distribution as the training set, and only use examples of groups **G1** and **G4**. Note that Tab. 1 refers to **G3** as the "Worst Group".

Training details: We train starting from models pre-trained on ImageNet [S13]. We fine-tune with fixed learning rate of 10^{-5} with λ_{loc} of 5×10^{-2} ($5 \times 10^{-4} \times 100$ for using 1% of annotations) using an Adam optimizer [S6] . We train for 350

[4] https://github.com/pytorch/pytorch

[5] https://github.com/pytorch/vision

[6] https://github.com/B-cos/B-cos-v2

[7] https://github.com/visinf/fast-axiomatic-attribution

[8] https://github.com/stevenstalder/NN-Explainer

[9] https://github.com/pytorch/captum

Localization Loss	**Values of λ_{loc}**
Energy	5×10^{-4}, 1×10^{-3}, 5×10^{-3}
L_1	1×10^{-3}, 5×10^{-3}, 1×10^{-2}
PPCE	1×10^{-4}, 5×10^{-4}, 1×10^{-3}
RRR*	5×10^{-6}, 1×10^{-5}, 5×10^{-5}

Table D1: **Hyperparameter λ_{loc}: Default training.** used for when training on VOC and COCO with each localization loss. Different values are used for different loss functions since the magnitudes of each loss varies.

Localization Loss	**Values of λ_{loc}**
Energy	0.05, 0.100, 0.50
L_1	0.01, 0.100, 1.00

Table D2: **Hyperparameter λ_{loc}: Limited annotations.** used for when training on VOC and COCO with **limited data** for each localization loss. Different values are used for different loss functions since the magnitudes of each loss varies. We use larger values of λ_{loc} when training with limited annotations to maintain the relative magnitudes of the classification and localization losses during training.

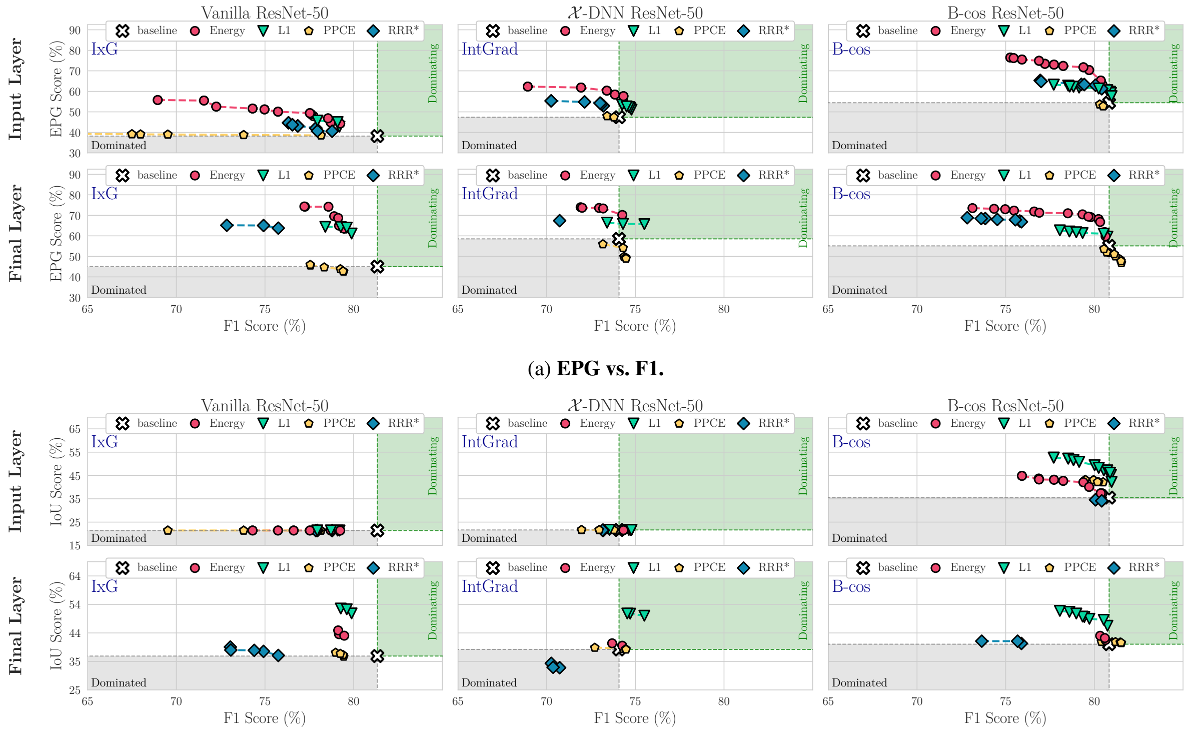
epochs with random cropping and horizontal flipping and select the checkpoint with the highest accuracy on the modified validation set.

D.2. Optimizing B-cos Attributions

Training for optimizing the localization of attributions (Eq. (1)) requires backpropagating through the attribution maps, which implies that they need to be differentiable. While B-cos attributions [S1] as formulated are mathematically differentiable, the original implementation[6] [S2] for computing them involves detaching the dynamic weights from the computational graph, which prevents them from being used for optimization. In this work, to use them for model guidance, we develop a twice-differentiable implementation of B-cos attributions.

X. Full Results

Full results on **PASCAL VOC 2007** (F1 score).

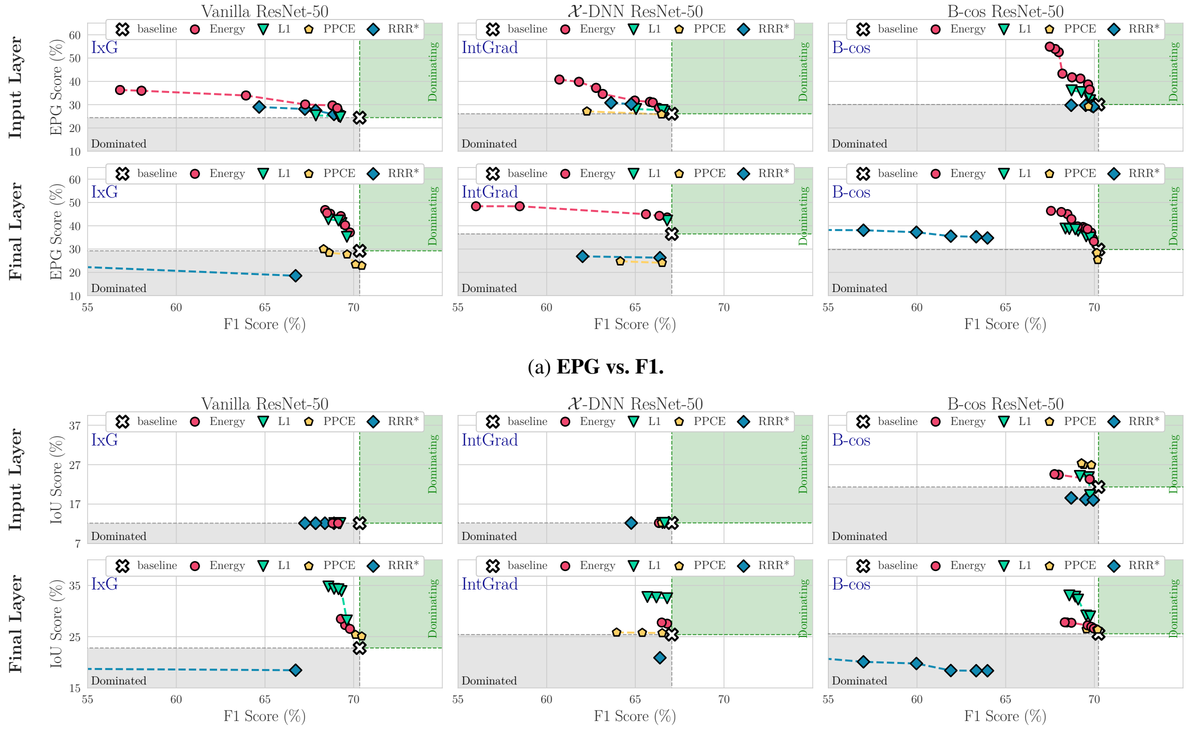


(a) **EPG vs. F1.**

(b) **IoU vs. F1.**

Fig. X1: **EPG (a) and IoU (b) vs. F1 on VOC,** for different losses (**markers**) and models (**columns**), optimized at different layers (**rows**); additionally, we show the performance of the baseline model before fine-tuning and demarcate regions that strictly dominate (are strictly dominated by) the baseline performance in green (grey). For each configuration, we show the Pareto fronts (cf. Fig. 4) across regularization strengths λ_{loc} and epochs (cf. Sec. 5 and Fig. 4). We find the Energy loss to give the best trade-off between EPG and F1, whereas the L_1 loss (especially at the final layer) provides the best trade-off between IoU and F1. We further find these results to be consistent across datasets, see Fig. X2.

Full results on **MS COCO 2014** (F1 score).

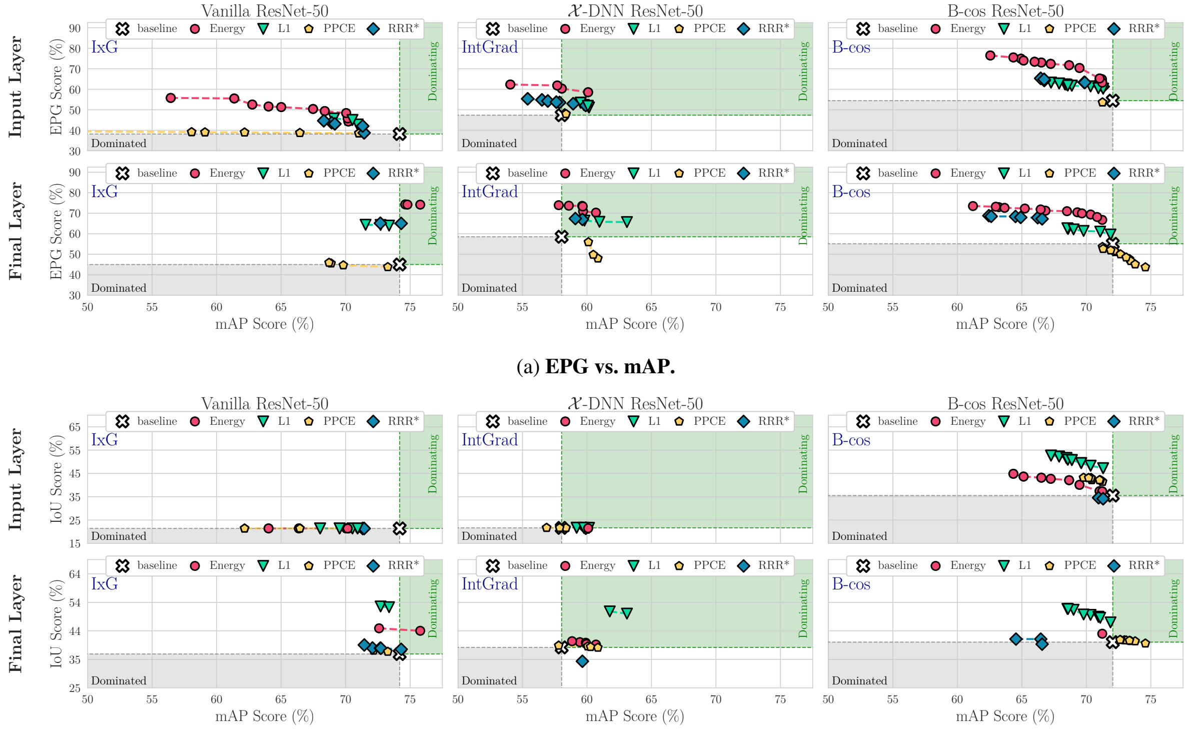


(a) **EPG vs. F1.**

(b) **IoU vs. F1.**

Fig. X2: **EPG (a) and IoU (b) vs. F1 on COCO,** for different losses (**markers**) and models (**columns**), optimized at different layers (**rows**); additionally, we show the performance of the baseline model before fine-tuning and demarcate regions that strictly dominate (are strictly dominated by) the baseline performance in green (grey). For each configuration, we show the Pareto fronts (cf. Fig. 4) across regularization strengths λ_{loc} and epochs (cf. Sec. 5 and Fig. 4). We find the Energy loss to give the best trade-off between EPG and F1, whereas the L_1 loss (especially at the final layer) provides the best trade-off between IoU and F1. We further find these results to be consistent across datasets, see Fig. X1.

Mean Average Precision (mAP) results on VOC.

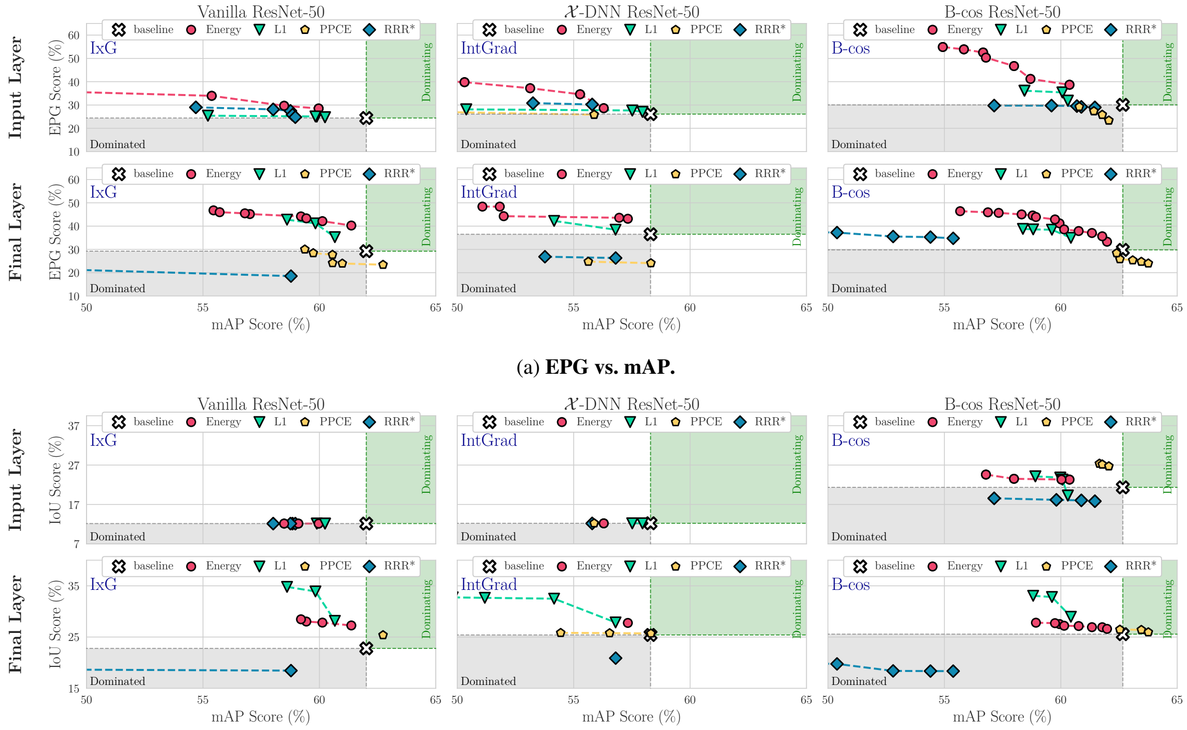


(a) **EPG vs. mAP.**

(b) **IoU vs. mAP.**

Fig. X3: **Quantitative comparison of EPG and IoU vs. mAP scores for VOC.** To ensure that the trends observed and described in the main paper generalize beyond the F1 metric, in this figure we show the EPG and IoU scores plotted against the mAP metric. In general, we find the results obtained for the mAP metric to be highly consistent with the previously shown results for the F1 metric, see Fig. X1. E.g., across all configurations, we find the Energy to yield the highest gains in EPG score, whereas the L_1 loss provides the best trade-offs with respect to the IoU metric. These results are further also consistent with those observed on COCO, see Fig. X4.

Full results on **MS COCO 2014** (mAP).

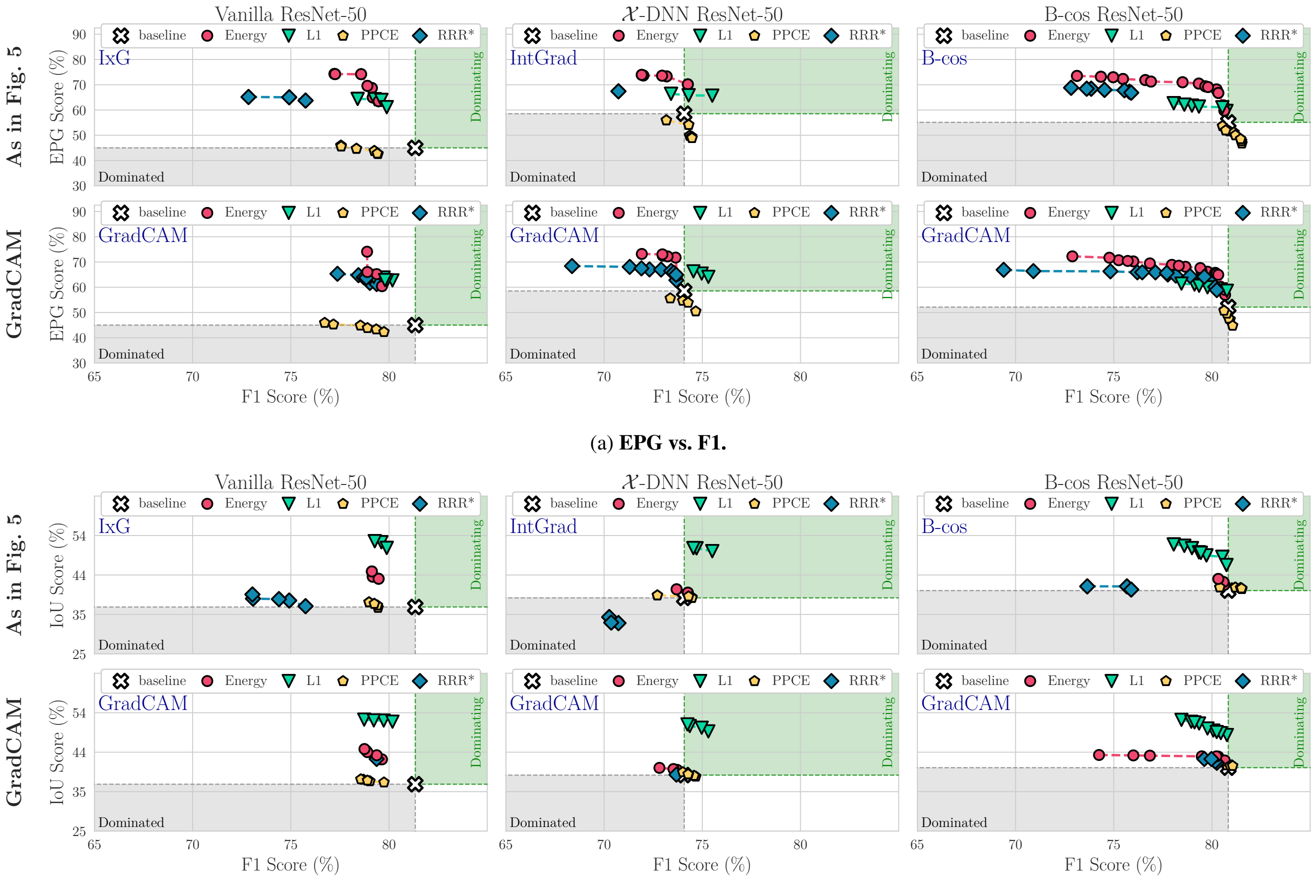


(a) **EPG vs. mAP.**

(b) **IoU vs. mAP.**

Fig. X4: **Quantitative comparison of EPG and IoU vs. mAP scores for COCO.** To ensure that the trends observed and described in the main paper generalize beyond the F1 metric, in this figure we show the EPG and IoU scores plotted against the mAP metric. In general, we find the results obtained for the mAP metric to be highly consistent with the previously shown results for the F1 metric, see Fig. X2. E.g., across all configurations, we find the Energy to yield the highest gains in EPG score, whereas the L_1 loss provides the best trade-offs with respect to the IoU metric. These results are further also consistent with those observed on VOC, see Fig. X3.

Comparison to GradCAM on VOC.

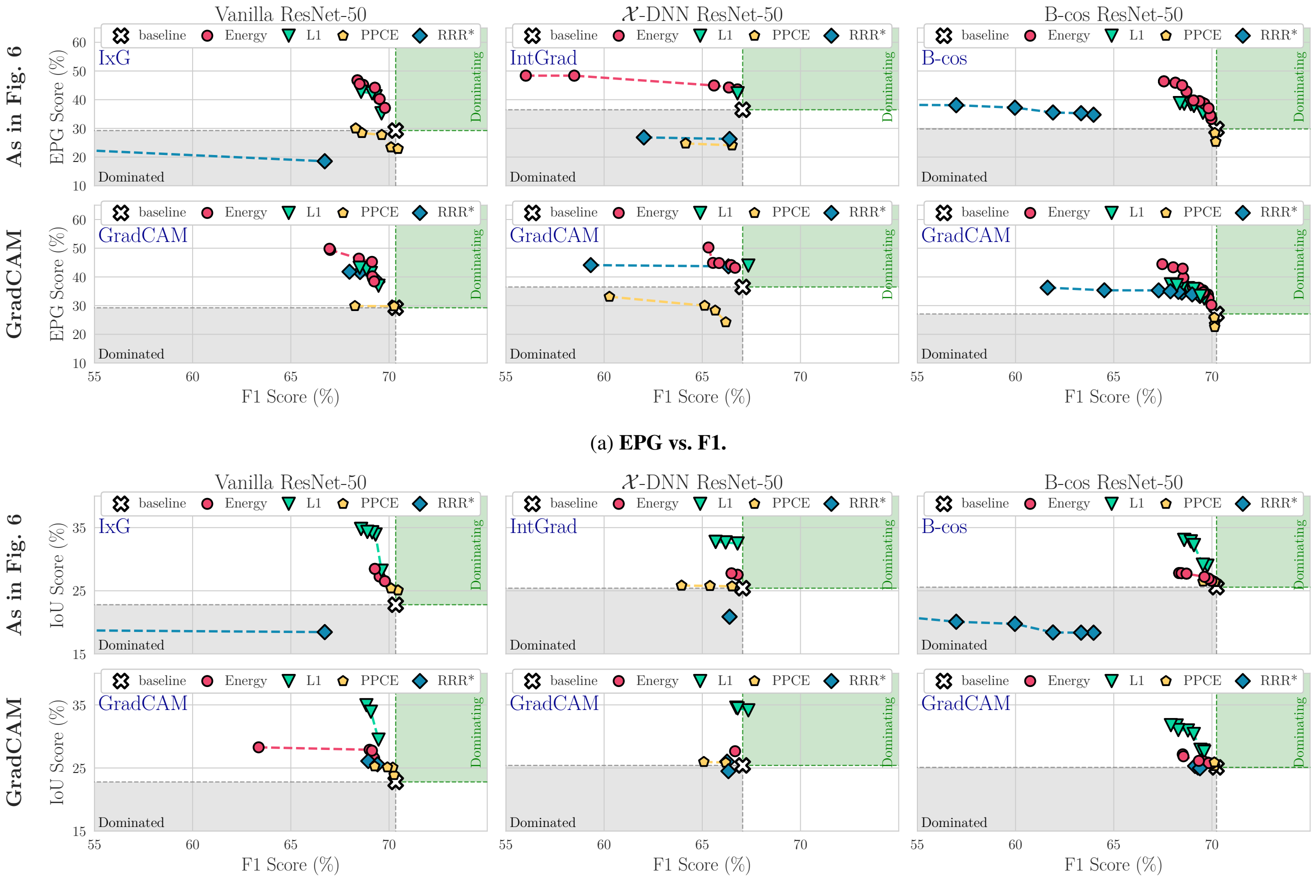


(a) **EPG vs. F1.**

(b) **IoU vs. F1.**

Fig. X5: **Quantitative results using GradCAM on VOC.** We show EPG **(a)** and IoU **(b)** scores vs. F1 scores for all localization losses and models using GradCAM at the final layer (**bottom rows** in (a)+(b) and compare it to the results shown in the main paper (**top rows**). As expected, GradCAM performs very similarly to IxG (Vanilla) and IntGrad ($\mathcal{X}$-DNN) used at the final layer—in particular, note that for ResNet-50 architectures, IxG and IntGrad are very similar to GradCAM for Vanilla and $\mathcal{X}$-DNN models respectively (see Sec. B.2). Similarly, we find GradCAM to also perform comparably to the B-cos explanations when used at the final layer; for results on COCO, see Fig. X6.

Comparison to GradCAM on COCO.



(a) **EPG vs. F1.**

(b) **IoU vs. F1.**

Fig. X6: **Quantitative results using GradCAM on COCO.** We show EPG **(a)** and IoU **(b)** scores vs. F1 scores for all localization losses and models using GradCAM at the final layer (**bottom rows** in (a)+(b) and compare it to the results shown in the main paper (**top rows**). As expected, GradCAM performs very similarly to IxG (Vanilla) and IntGrad ($\mathcal{X}$-DNN) used at the final layer—in particular, note that for ResNet-50 architectures, IxG and IntGrad are very similar to GradCAM for Vanilla and $\mathcal{X}$-DNN models respectively (see Sec. B.2). Similarly, we find GradCAM to also perform comparably to the B-cos explanations when used at the final layer; for results on VOC, see Fig. X5.

EPG results for **intermediate layers** on VOC.

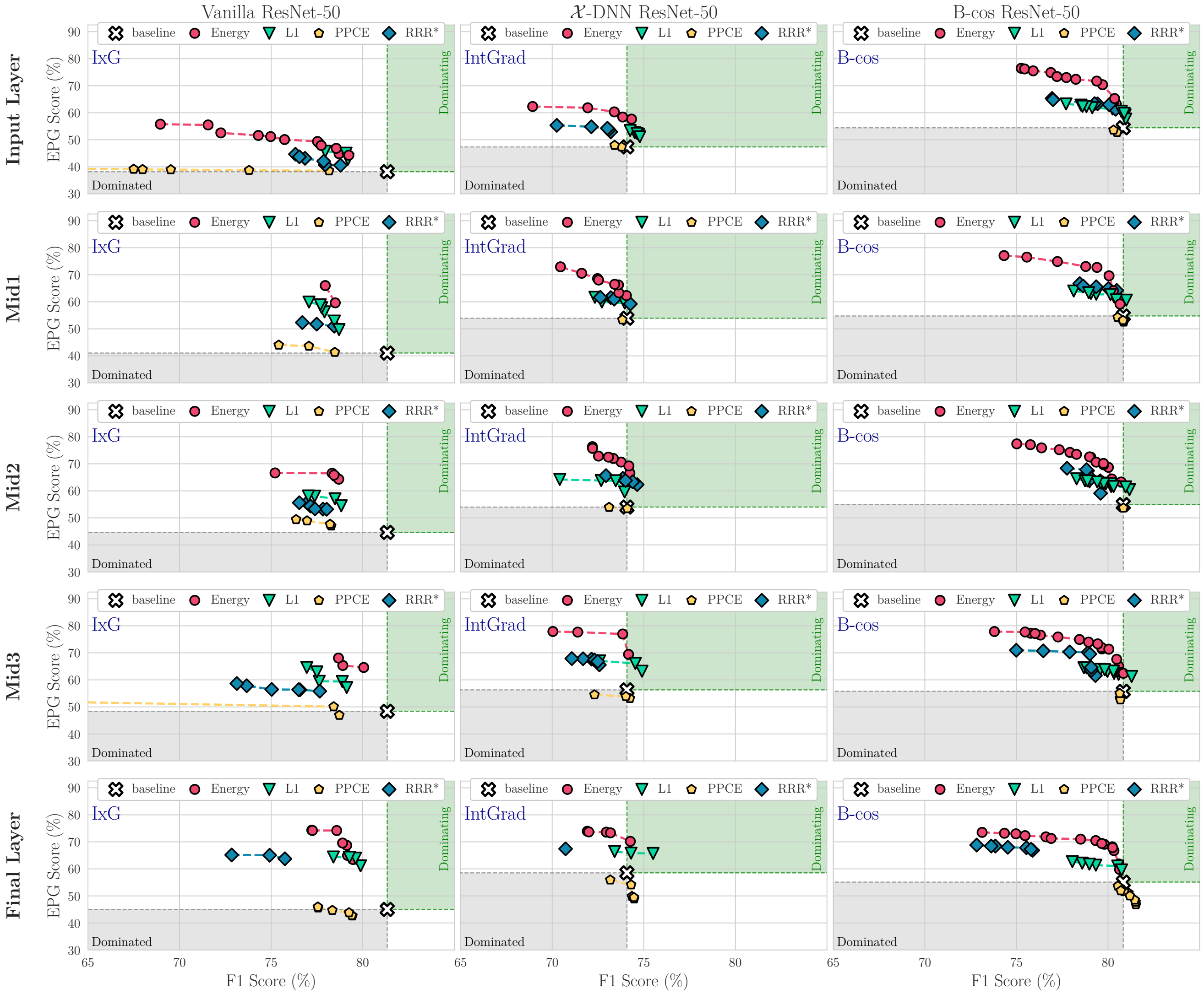


Fig. X7: **Intermediate layer results comparing EPG vs. F1.** We compare the effectiveness of model guidance at varying network depths (**rows**) for each attribution method and model (**columns**) across localization loss functions. For the B-cos model, we find similar trends at all network depths, with the Energy localization loss outperforming all other losses. For the Vanilla and $\mathcal{X}$-DNN models, the Energy loss similarly performs the best, but we also observe improved performance across losses when optimizing at deeper layers of the network. Results for IoU can be found in Fig. X8.

IoU results for **intermediate layers** on VOC.

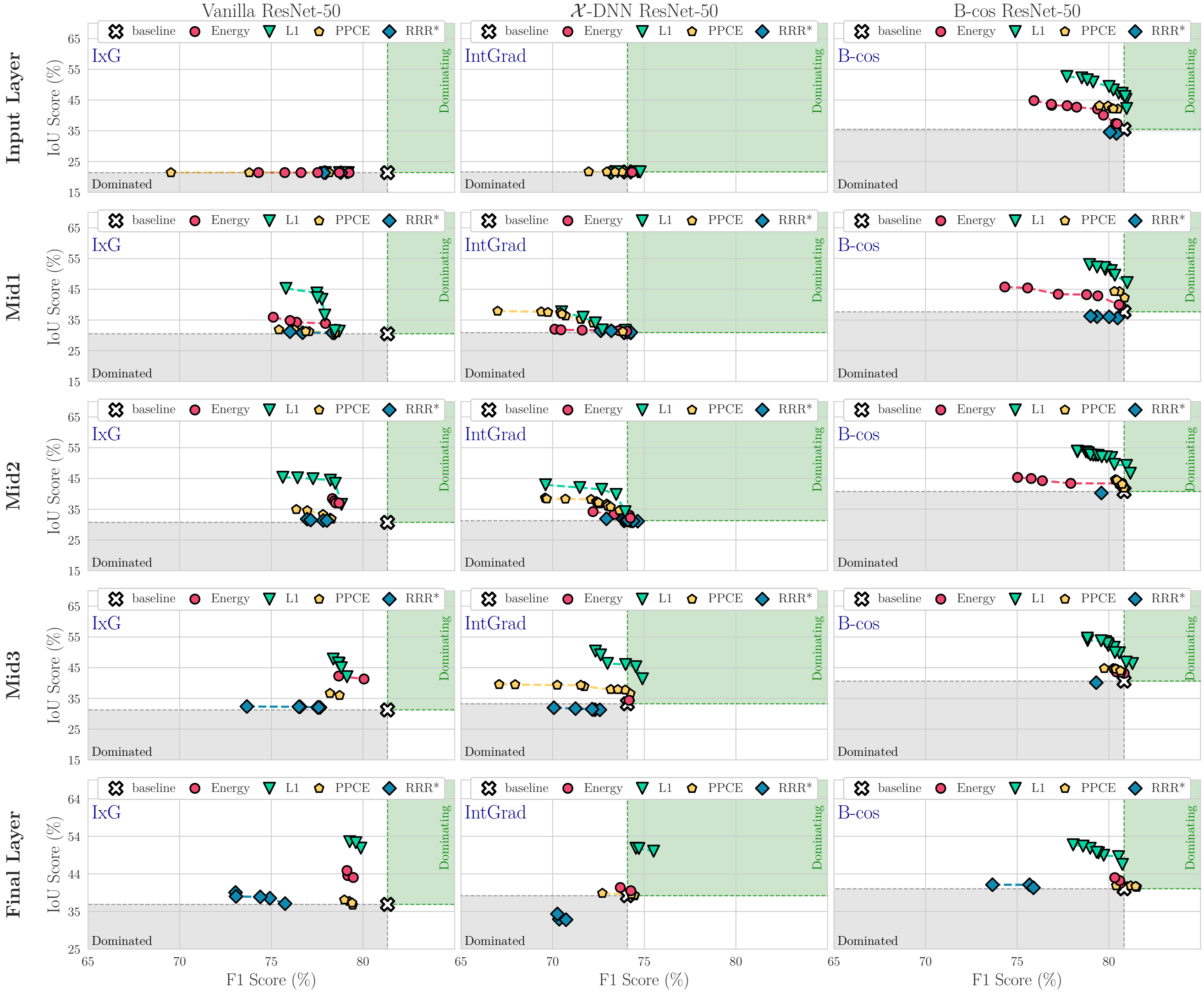
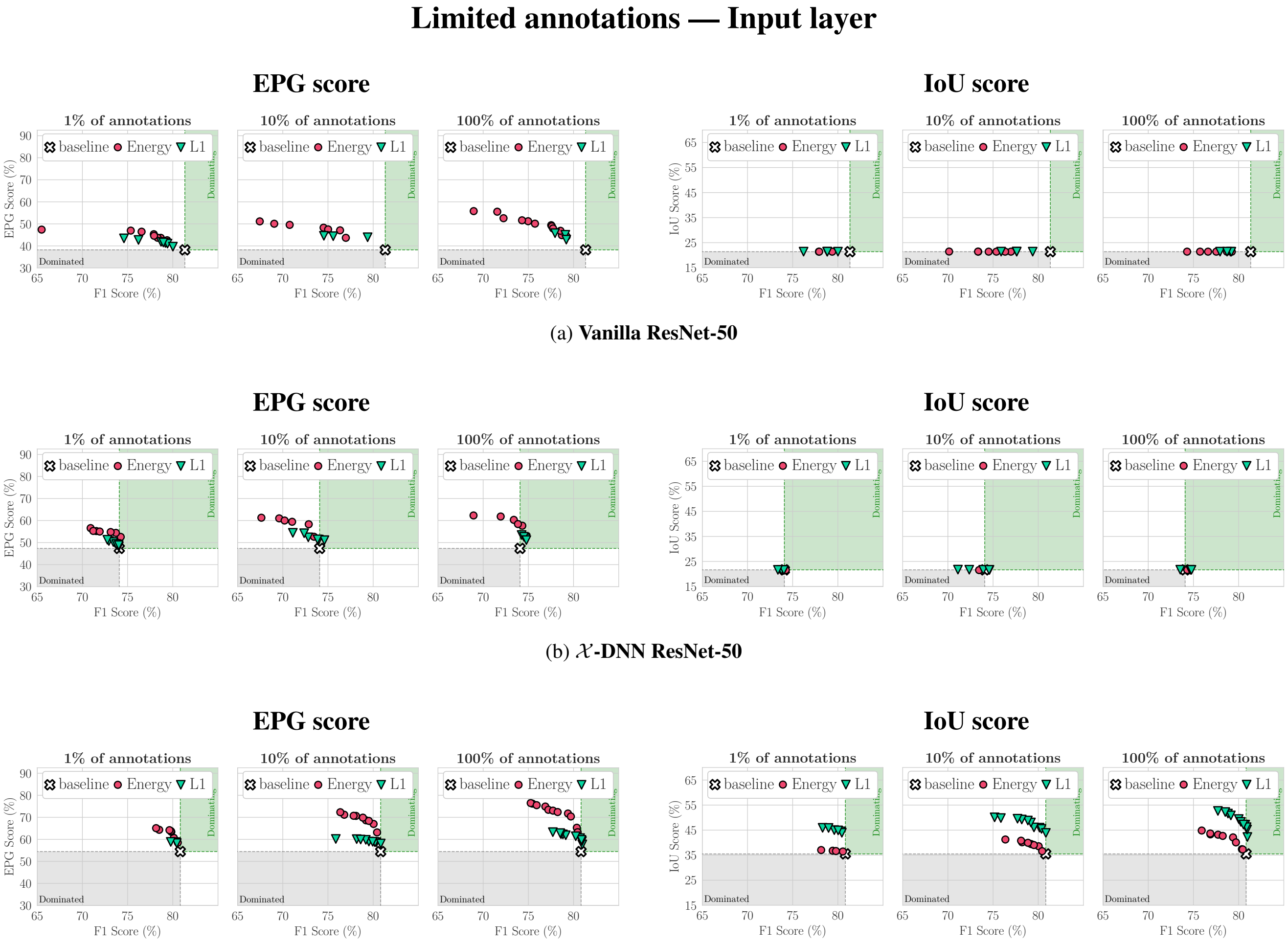


Fig. X8: **Intermediate layer results comparing IoU vs. F1.** We compare the effectiveness of model guidance at varying network depths (**rows**) for each attribution method and model (**columns**) across localization loss functions. We find similar trends across all configurations, with the L_1 loss outperforming all other losses. For the Vanilla and $\mathcal{X}$-DNN models, we observe improved performance across losses when optimizing at deeper layers of the network, whereas the results seem very stable for the B-cos models. For EPG results, see Fig. X7.

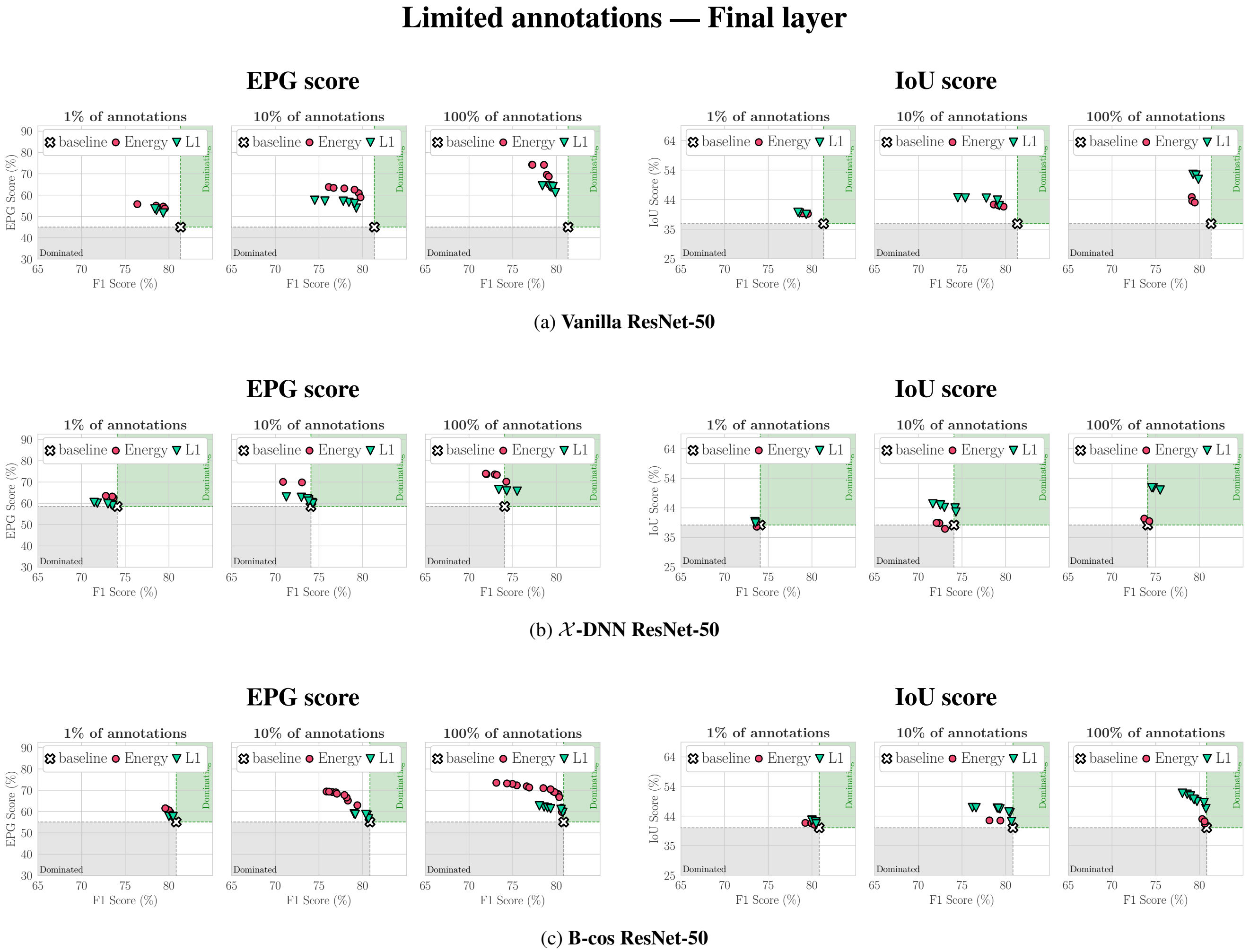


(a) **Vanilla ResNet-50**

(b) $\mathcal{X}$**-DNN ResNet-50**

(c) **B-cos ResNet-50**

Fig. X9: **EPG and IoU scores for model guidance at the input layer with a limited number of annotations.** We show EPG vs. F1 (**left**) and IoU vs. F1 (**right**) for all models, optimized with the Energy and L_1 localization losses, when using $\{1\%, 10\%, 100\%\}$ training annotations. We find that model guidance is generally effective even when training with annotations for a limited number of images. While the performance slightly worsens when using 1% annotations, using just 10% annotated images yields similar gains to using a fully annotated training set. Results at the final layer can be found in Fig. X10.



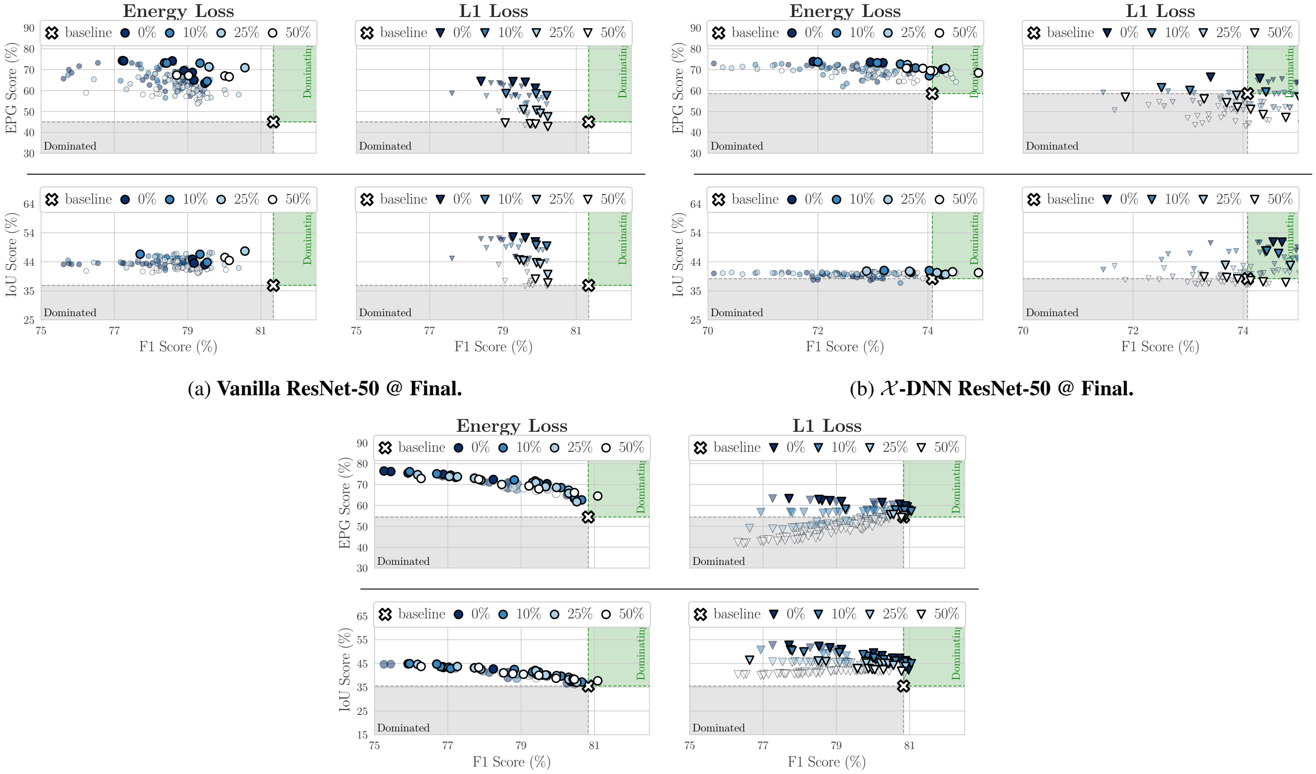
(a) **Vanilla ResNet-50**

(b) $\mathcal{X}$**-DNN ResNet-50**

(c) **B-cos ResNet-50**

Fig. X10: **EPG and IoU scores for model guidance at the final layer with a limited number of annotations.** We show EPG vs. F1 (**left**) and IoU vs. F1 (**right**) for all models, optimized with the Energy and L_1 localization losses, when using $\{1\%, 10\%, 100\%\}$ training annotations. We find that model guidance is generally effective even when training with annotations for a limited number of images. While the performance worsens when using 1% annotations, using just 10% annotated images yields similar gains to using a fully annotated training set. Results at the input layer can be found in Fig. X9.

Additional results for training with **coarse bounding boxes**



(a) **Vanilla ResNet-50 @ Final.**

(b) $\mathcal{X}$**-DNN ResNet-50 @ Final.**

(c) **B-cos ResNet-50 @ Input.**

Fig. X11: **Coarse bounding box results**. We show the impact of dilating bounding boxes during training for the **(a)** Vanilla and **(b)** $\mathcal{X}$-DNN, and **(c)** B-cos models. Similar to the results seen with B-cos models (c), we find that the Energy localization loss is generally robust to coarse annotations, while the effectiveness of guidance with the L_1 localization loss worsens as the extent of coarseness increases.

References for Supplement

[S1] Moritz Böhle, Mario Fritz, and Bernt Schiele. B-Cos Networks: Alignment is All We Need for Interpretability. In *CVPR*, pages 10329–10338, 2022.

[S2] Moritz Böhle, Navdeeppal Singh, Mario Fritz, and Bernt Schiele. B-cos Alignment for Inherently Interpretable CNNs and Vision Transformers. *arXiv preprint arXiv:2306.10898*, 2023.

[S3] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *IJCV*, 88:303–308, 2009.

[S4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, pages 770–778, 2016.

[S5] Robin Hesse, Simone Schaub-Meyer, and Stefan Roth. Fast Axiomatic Attribution for Neural Networks. In *NeurIPS*, pages 19513–19524, 2021.

[S6] Diederik P Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *ICLR*, 2015.

[S7] Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, and Orion Reblitz-Richardson. Captum: A Unified and Generic Model Interpretability Library for PyTorch. *arXiv preprint arXiv:2009.07896*, 2020.

[S8] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *ECCV*, pages 740–755. Springer, 2014.

[S9] TorchVision Maintainers and Contributors. TorchVision: PyTorch's Computer Vision Library. https://github.com/pytorch/vision, 2016.

[S10] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary 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 *NeurIPS*, 2019.

[S11] Suzanne Petryk, Lisa Dunlap, Keyan Nasseri, Joseph Gonzalez, Trevor Darrell, and Anna Rohrbach. On Guiding Visual Attention with Language Specification. In *CVPR*, pages 18092–18102, 2022.

[S12] Sukrut Rao, Moritz Böhle, and Bernt Schiele. Towards Better Understanding Attribution Methods. In *CVPR*, pages 10223–10232, 2022.

[S13] 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. *IJCV*, 115(3):211–252, 2015.

[S14] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally Robust Neural Networks for Group Shifts: On the Importance of Regularization for Worst-Case Generalization. In *ICLR*, 2020.

[S15] 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 *ICCV*, pages 618–626, 2017.

[S16] Steven Stalder, Nathanaël Perraudin, Radhakrishna Achanta, Fernando Perez-Cruz, and Michele Volpi. What You See is What You Classify: Black Box Attributions. In *NeurIPS*, 2022.