Towards Better Guided Attention and Human Knowledge Insertion in Deep Convolutional Neural Networks

Anonymous ECCV BIC submission

Paper ID 6

Abstract. Attention Branch Networks (ABNs) have been shown to simultaneously provide visual explanation and improve the performance of deep convolutional neural networks (CNNs). In this work, we introduce Multi-Scale Attention Branch Networks (MSABN), which enhance the resolution of the generated attention maps, and improve the performance. We evaluate MSABN on benchmark image recognition and fine-grained recognition datasets where we observe MSABN outperforms ABN and baseline models. We also introduce a new data augmentation strategy utilizing the attention maps to incorporate human knowledge in the form of bounding box annotations of the objects of interest. We show that even with a limited number of edited samples, a significant performance gain can be achieved with this strategy.

Keywords: visual explanation, fine-grained recognition, attention map, human-in-the-loop

1 Introduction

CNNs have established themselves as the benchmark approach in image recognition [13, 20, 33, 30]. However, the interpretation of the decision-making process still remains elusive. To be able to visualize and verify that the decision of a CNN is based on correct and meaningful information is important for many computer vision applications like self-driving cars or automated social media content analysis, to increase reliability and trust in the system. For biomedical applications, where instrument settings and small cohorts/datasets easily bias the results this becomes extremely important. This, in addition to the high cost or risk associated with erroneous decisions, limit the reliability and hence also the deployment of CNN-based solutions in clinical diagnostics and biomedical analysis. Tools to explain the decision and be able to correct erroneous conclusions drawn by CNN would improve the trustworthiness of the technology.

Visual explanation [35] is used in deep learning to interpret the decisions of the CNNs. Broadly, these methods can be categorized as either requiring additional backpropagation or not. Methods requiring backpropagation [27, 5] can be used out-of-the-box and don't require any network architectural or training method changes. However, they need an extra backpropagation step to find the

discriminative regions in images. Response-based methods [36, 10] don't require backpropagation because they generate explanation and predictions simultaneously. In this work, we improve upon [10] and present a response-based visual explanation method that outperforms previous methods.

To fully utilize the benefits of deep learning in practical settings, it would also be beneficial to be able to incorporate human knowledge and interact with the attention maps to correct and improve the networks. One attempt to do this is presented in [22], where the attention map of the images is manually edited and then the network is fine-tuned with this human knowledge. This approach, however, requires manual detailed annotation (drawing) of the attention maps which becomes tedious for large datasets. Here, we propose a softer form of user-input in the form of object bounding boxes.

In this paper we build upon the attention branch network structure and present ways to improve the attention maps and overall network performance even further. In addition, we present a simple and efficient way to incorporate human expertise when training/refining the networks. Figure 1 illustrates the difference in the attention map detail achieved with our proposed multi-scale attention branch network (MSABN) compared to the attention branch network (ABN) and the commonly used Class Activation Mapping (CAM).

Our main contributions are:

- With the introduction of MSABN, we increase the performance of the response-based guided attention models significantly and simultaneously increase the resolution of attention maps by 4x which has not been achieved with the response-based methods yet.
- With the introduction of the puzzle module in the attention branch, the performance is improved further and it also adds better localization performance for fine-grained recognition.
- We provide a human-in-the-loop (HITL) pipeline for inserting human knowledge in the models which require simple annotation (object bounding box) and achieve performance improvement even by annotating/correcting a small portion of the dataset.

2   Related Work

Visual explanation: Highlighting the regions or details of an image that are important for the decision is commonly used to interpret and verify the decision-making process of CNNs [27, 2]. Overall, the methods for visual explanation fall in two categories: gradient-based and response-based. Gradient-based methods (SmoothGrad [28], Guided Backpropagation [29], LIME [25], grad-CAM [27], gradCAM++ [5], LRP [2]) rely on the backpropagation of auxiliary data like class index or noise to find the discriminative regions in the image. These methods do not require training or architectural modifications. However, backpropagation adds another step during inference and increases the computational cost. Response-based methods, on the other hand, don not require backpropagation
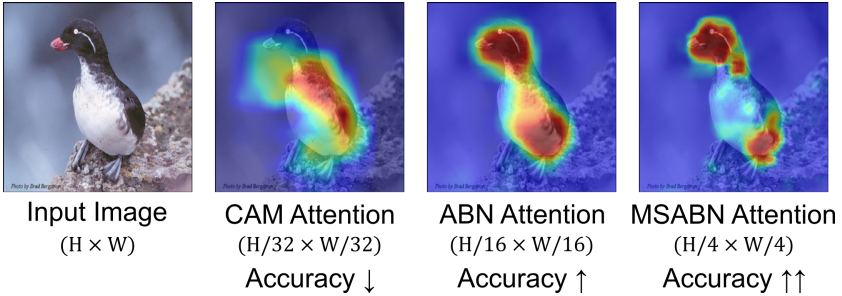
Input Image	CAM Attention	ABN Attention	MSABN Attention
(H × W)	(H/32 × W/32)	(H/16 × W/16)	(H/4 × W/4)
	Accuracy ↓	Accuracy ↑	Accuracy ↑↑

Fig. 1. Comparison of the MSABN attention map with CAM and ABN attention outputs. MSABN provides better accuracy and higher resolution attention maps.

and can provide interpretation at the same time as inference. CAM [36], is a response-based method which provides an attention map for each class by replacing the final fully-connected layer of a CNN with a convolutional layer and global average pooling. The recently proposed attention branch networks [10] extend the response-based model by adding a branch structure to provide the attention to the network in a feed-forward way and show that adding attention in this way improves the performance of the models.

Inserting human knowledge: Human-in-the-loop (HITL) refer to approaches in the deep learning framework where a prediction model is trained with integrated human knowledge or user guidance. Several HITL approaches [4, 3, 32, 24, 23] have been used to inject human knowledge in different ways to the deep neural networks. In [21], the ClickMe interface is used to get human annotation in the form of mouse clicks to get approximate attention regions in an image. They also modified the model architecture to incorporate this human knowledge and demonstrated improved performance. In [26], the authors proposed a method that leverages the explanation generated by the models such that they should provide the correct explanation for the correct prediction. In [22], Mitsuhara, M. et al. showed that editing the attention maps and sequentially fine-tuning the ABN models improve performance as mentioned in the Introduction. However, their approach involve manually marking the object boundary in a large number (thousands) of images making it unfeasible in practice. In [15, 1], the authors attempted to improve model performance by using non-strict attention input and introduce different loss functions and manage to improve performance. In [15], Dharma et al. use different loss functions for the regions inside and outside the object bounding box as a way to incorporate human input. Our method is similar to these methods but focused on augmenting the data rather than changing the loss function.

Copy-paste augmentation: This refers to an augmentation strategy where a patch from one image is pasted into another image which is then used for train-

ing. In [7, 8, 11] this augmentation strategy is used to improve the performance in instance segmentation and detection tasks. The idea is that by cutting object instances and pasting them into other images a different context for the objects is provided. The CutMix [34] augmentation strategy used this idea more generally and show that simply copying and pasting random patches from one image to another and modifying the output probabilities improves both the classification and localization performance. Finally, PuzzleMix [16] explored CutMix combined with saliency information and image statistics to further improve the performance. In the augmentation method we propose, the way to incorporate human knowledge can be seen as a supervised version of CutMix. We use the object location to copy and replace the objects in the dataset in the images with mismatched attention map and true object location.

3   Methods

3.1   Multi-Scale Attention Branch Network

In [10], a dual attention and perception branch architecture is used to generate the attention map and prediction simultaneously. The output from the third convolutional block (which is referred to as "feature extractor") is fed into the attention branch which produces the CAM output and the attention map. The attention map is then combined with the feature extractor output and fed into the perception branch which outputs the probability of each class. The CAM output from the attention branch and the probability output of the perception branch are trained on cross-entropy loss simultaneously. The training can then be done in an end-to-end manner in both branches. The loss can be written as,

$$\mathcal{L}(x_i) = \mathcal{L}_{attn}(x_i) + \mathcal{L}_{cls}(x_i) \qquad (1)$$

where $\mathcal{L}_{attn}(x_i)$ and $\mathcal{L}_{cls}(x_i)$ are the cross-entropy losses for the attention and perception branch respectively for a training sample x_i.

Unlike [10], where the attention branch is only fed the output of the third convolutional block, the attention branch of our multi-scale attention branch network (MSABN) is fed with input from the first and second convolutional blocks as well. Figure 2 shows the MSABN architecture. The output from the second and third convolutional blocks are upsampled to match the size of the first convolutional block. The output from each of the three convolutional blocks is then passed through 1×1 convolutional blocks such that the total number of channels after concatenating the outputs match the input channels of the attention branch. Our suggested architecture helps the network accumulate hierarchical information at different scales to produce a fine-grained attention map and improve performance. The resolution of the attention map in MSABN is also higher than in ABN as a result of the upsampling.
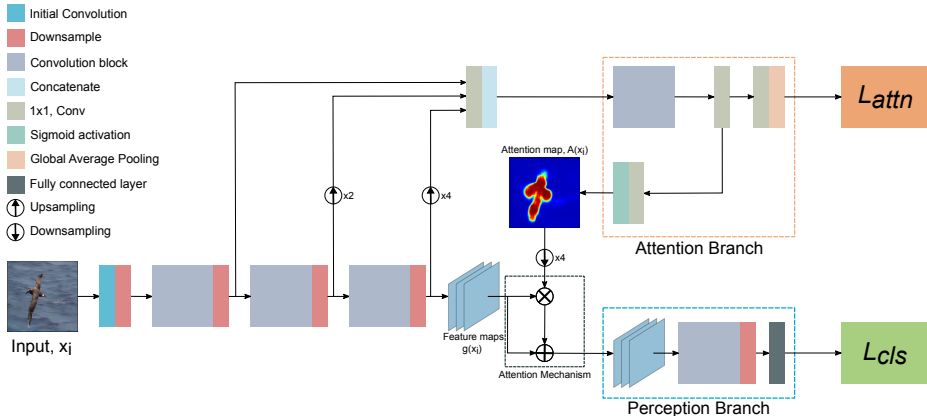
Fig. 2. Overview of our proposed MSABN framework.

3.2 Puzzle Module to Improve Fine-grained Recognition

The puzzle module consists of a tiling and a merging module which aims at minimizing the difference between the merged features from tiled patches in an image and the features from the original image. The tiling module generates non-overlapping tiled patches of the input image. The patches are fed into the network and the merging module merges the patches' outputs into CAMs. The L1 loss between the CAM of the target class from the original image and the reconstruction is then calculated and used for training. Figure 3 illustrates the puzzle module. The overall loss for training MSABN models with the puzzle module becomes:

$$\mathcal{L}(x_i) = \mathcal{L}_{attn}(x_i) + \mathcal{L}_{cls}(x_i) + \mathcal{L}_{re}(x_i), \tag{2}$$

where $\mathcal{L}_{re}(x_i)$ is the L1 loss between the original image CAM and the reconstructed CAM.

The puzzle module was introduced in [14] for weakly-supervised segmentation. It was shown to improve the segmentation performance on the PASCAL VOC 2012 dataset [9] with CAM based ResNet models. Since the attention branch is trained similar to CAM models, we decided to applied the puzzle module to the attention branch to observe the effects on attention maps and the performance of the models.

3.3 Embedding Human Knowledge with Copy-Replace Augmentation

We describe a HITL pipeline with the proposed fine-tuning process incorporating human knowledge. First, the MSABN model is trained with images in the training set with corresponding labels. Then, the attention maps of the training samples are obtained from the best performing model. In a real setting, the user
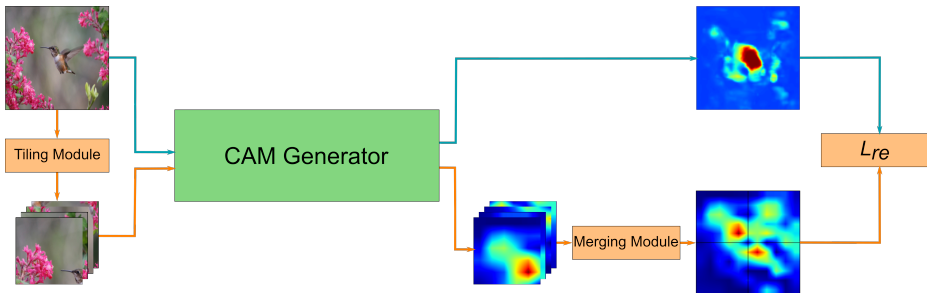
Fig. 3. Overview of the Puzzle Module. The tiling module divides the image into non-overlapping tiles which are then fed into the CAM generator and the merging module merges the tiled CAMs that is then compared with the original CAM.

can then inspect the attention maps and provide corrective input where the attention is mislocated in the form of a bounding box around the object. This is a softer form of user input compared to object boundary annotation and can be scaled to some degree without it becoming too cumbersome. Then, the training images are divided into two pools, one with bounding box annotations provided by the user and the other without. In the fine-tuning step, we train with all the training images and labels normally, *but* we modify the images where the bounding box annotation was provided. For the images with provided annotations, the patch defined by the bounding box in an image is extracted, resized, and pasted onto the bounding box of another image from the same pool as illustrated in Figure 4. The resizing to the target bounding box size also acts as scaling augmentation during the training.

4   Experiments

4.1   Image Classification

Datasets: We evaluate MSABN for image classification on the CIFAR100 [19], ImageNet [6] and DiagSet-10x [17] datasets. CIFAR100 contains 60,000 images for training and 10,000 images for testing of 100 classes with image size of 32×32 pixels. ImageNet consists of 1,281,167 training images and 50,000 validation images of 1000 classes. Finally, the DiagSet-10x dataset is a histopathology dataset for prostate cancer detection with 256×256 pixel images divided into 9 categories with different cancer grading, normal tissue, artifacts, and background. This dataset was included to test the performance of MSABN on texture data rather than object data. The dataset consists of 132,882 images for training, 25,294 for validation, and 30,086 for testing. In the experiments, the background class (empty, non-tissue regions of slide) was excluded.

Training details: For the CIFAR100 dataset, ResNet-20,-56,-110 [13] models were used to evaluate the performance. The models were optimized with stochas-
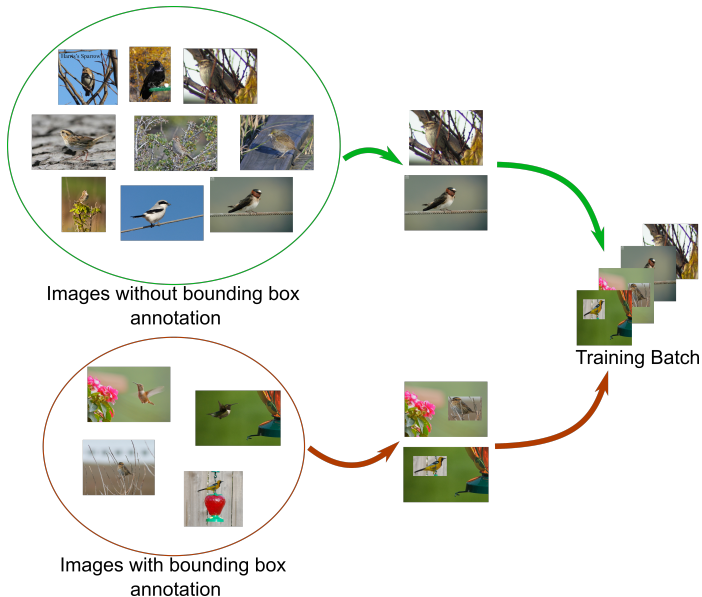
Fig. 4. Overview of the copy-paste training scheme. In our case, objects with misaligned attention are annotated with a bounding box and, cut out, and replaced by other bounding box annotated objects.

tic gradient descent (SGD) with a momentum of 0.9 and weight decay of 5e-4 for 300 epochs and a batch size of 256. The initial learning rate was set to 0.1 and was divided by 10 at 50% and 75% of the total number of epochs. Random cropping and horizontal flipping were used as augmentations, and model weights were initialized with Kaiming initialization [12]. The experiments were repeated three times with different random seeds and the mean and standard deviation of the accuracy is reported (mainly to assert stability of the models).

For the DiagSet-10x and ImageNet datasets, ResNet-101 and EfficientNet-B1 models were used. The models were optimized with SGD with a momentum of 0.9 and weight decay of 1e-4 for 90 epochs. A batch size of 512 was used for ResNet-101 experiments and of 256 for EfficientNet-B1 experiments. The initial learning rate was set to 0.1 and was divided by 10 at 33% and 66% of the total number of epochs. For both datasets, random cropping of 224×224 and horizontal flipping was used as augmentations. For DiagSet-10x vertical flipping and color jitter were also used. Due to severe imbalance between classes in the DiagSet-10x dataset, the loss was weighted by the number of samples of each class. All model weights were initialized with Kaiming initialization. These experiments were only performed once due to computational constraints. EfficientNet-B1+ABN performance was not reported as the authors in [10] didn't implement ABN on EfficientNets.

Analysis of attention mechanism on CIFAR: Following the analysis in [10], two different attention mechanisms (how to combine the attention map with the feature maps), namely $g(x) \cdot A(x)$ and $g(x) \cdot (1 + A(x))$, were compared with the base (no attention branch) and ABN model which uses the $g(x) \cdot (1 + A(x))$ mechanism. As can be seen from Table 1, both mechanisms outperform the ABN for all model versions. $g(x) \cdot (1 + A(x))$ outperforms $g(x) \cdot A(x)$ slightly for the Resnet-20 and 110 models but performs slightly worse for ResNet-56 on average. The standard deviation for $g(x) \cdot A(x)$ mechanism is higher than for $g(x) \cdot (1 + A(x))$ for all models. This might be due to greater stability provided by the residual connection. We also noticed that the attention maps when using $g(x) \cdot A(x)$ have values close to 0.5 for the background as compared to $g(x) \cdot (1 + A(x))$ where they were close to zero. Hence we decided to use $g(x) \cdot (1 + A(x))$ mechanism as the default for this paper.

Table 1. Comparison of the accuracies (%) on CIFAR100

	ResNet-20	ResNet-56	ResNet-110
BaseModel	68.71±0.202	74.92±0.520	76.83±0.233
BaseModel+ABN	68.97±0.074	76.31±0.029	77.92±0.102
BaseModel+MSABN ($g(x) \cdot A(x)$)	69.47±0.298	**77.16±0.419**	78.40±0.254
BaseModel+MSABN ($g(x) \cdot (1 + A(x))$)	**70.06±0.025**	76.78±0.149	**78.68±0.121**

Accuracy on ImageNet and DiagSet-10x: Table 2 summarizes the results of the experiments on ImageNet. The MSABN model outperforms the ABN model by 0.35% for ResNet-101 and the base model by 0.76% and 1.41% for ResNet-101 and EfficientNet-B1 respectively. The visual comparison of the attention in ABN and MSABN models is shown in Figure 5. The MSABN attention has a better localization performance and highlights the object boundaries better than ABN attention.

Table 2. Comparison of the accuracies (%) on the ImageNet Dataset

	ResNet-101	EfficientNet-B1
BaseModel	77.33	67.61
BaseModel+ABN	77.74	-
BaseModel+MSABN	**78.09**	**69.08**

Table 3 summarizes the results of the experiments on DiagSet-10x. In addition to accuracy, we also report the balanced accuracy to better estimate the performance on an unbalanced dataset. Here, the MSABN model outperforms the ABN model by 4.23%/2.68% for ResNet-101 and the base model by 2.34%/2.72% and 0.09%/1.91% for ResNet-101 and EfficientNet-B1 respectively.
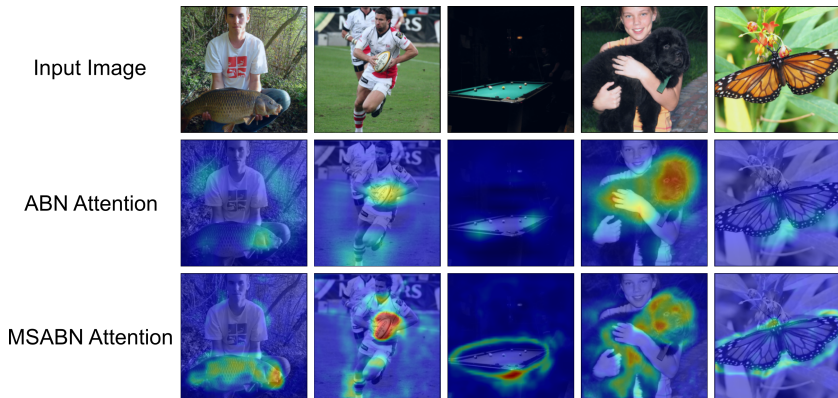
Fig. 5. Visualization of high attention regions in the images from the ImageNet dataset.

A visual comparison of the attention in ABN and MSABN models is shown in Figure 6. As compared to ABN, the MSABN attention highlights specific nuclei important for the grading. ABN attention maps also show an activation along the image boundary in the images with non-tissue regions present as an artifact. This artifact was not observed in the MSABN attention maps.

Table 3. Comparison of the accuracies (%) on the DiagSet-10x dataset

	ResNet-101 (Accuracy/Balanced Accuracy)	EfficientNet-B1 (Accuracy/Balanced Accuracy)
BaseModel	54.96 / 38.57	56.15 / 40.36
BaseModel+ABN	53.07 / 38.61	-
BaseModel+MSABN	**57.30 / 41.29**	**56.24 / 42.27**

4.2 Fine-grained Recognition

Datasets: The performance of MSABN and Puzzle-MSABN was evaluated in two fine-grained recognition datasets namely CUB-200-2011 [31], and Stanford Cars [18]. The CUB-200-2011 dataset contains 11788 images of 200 categories, 5994 for training and 5794 for testing. The Stanford Cars dataset contains 16185 images of 196 classes and is split into 8144 training images and 8041 testing images.

Training details: The models used for evaluation of the fine-grained datasets were ResNet-50 [13], ResNext-50 (32×4d) [33], and EfficientNet-B3 [30]. The models showcase different popular architectures and hence were chosen to benchmark the results. The models were optimized with SGD with a momentum of
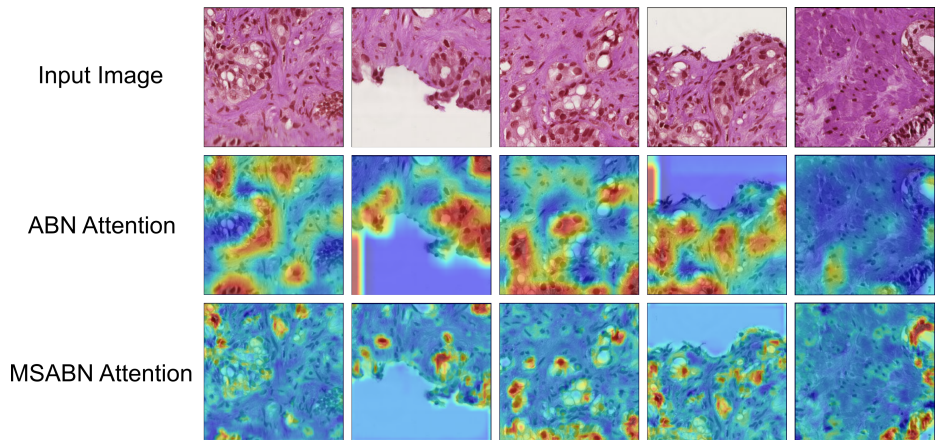
Fig. 6. Visualization of high attention regions in the DiagSet-10x dataset.

0.9 and weight decay of 1e-4 for 300 epochs with a batch size of 16. The initial learning rate was set to 0.1 and was divided by 10 at 50% and 75% of the total number of epochs. The images were resized to 352×352 pixels and the augmentations used were horizontal flip, color jitter, gaussian blur and noise, and solarize. All model weights were initialized with Kaiming initialisation. The experiments were run three times with different random seeds and the mean and standard deviation of the performance is reported.

Table 4 shows the accuracies of the models on the CUB-200-2011 dataset. MSABN improves upon the average performance of ABN models by 4% and 2% for ResNet-50 and ResNext-50 respectively. The performance gain compared with the base model is 14%, 12%, and 12% for ResNet-50, ResNext-50, and EfficientNet-B3 respectively. Furthermore, introduction of the puzzle module outperforms the ABN models by 6% and 4% respectively. The puzzle module outperforms the base models by 17%, 14%, and 13% for ResNet-50, ResNext-50, and EfficientNet-B3, respectively.

Table 4. Comparison of the accuracies (%) on CUB-200-2011 Dataset

	ResNet-50	ResNext-50	EfficientNet-B3
BaseModel	40.62±1.985	44.68±0.816	53.41±0.792
BaseModel+ABN	50.96±2.145	54.88±0.700	-
BaseModel+MSABN	54.98±0.219	56.86±0.581	65.28±0.397
BaseModel+MSABN+Puzzle	**57.31±0.645**	**58.87±0.592**	**66.21±0.615**

Table 5, shows the performance of the different models on the Stanford Cars dataset. The improvement with MSABN compared to the ABN model is not as significant here but slightly outperform them on average. The performance

gain compared with the base models however is 8%, 3%, and 2% for ResNet-50, ResNext-50, and EfficientNet-B3, respectively. The introduction of the puzzle module outperforms ABN models by 3% and 1% respectively. The puzzle module outperforms the base models by 10%, 4%, and 3% for ResNet-50, ResNext-50, and EfficientNet-B3, respectively.

Table 5. Comparison of the accuracies (%) on Stanford Cars Dataset

	ResNet-50	ResNext-50	EfficientNet-B3
BaseModel	78.70±1.062	84.00±0.817	86.86±0.495
BaseModel+ABN	85.59±0.396	87.34±0.408	-
BaseModel+MSABN	86.81±0.991	87.39±0.526	88.90±0.246
BaseModel+MSABN+Puzzle	**88.25±1.492**	**88.32±1.184**	**89.92±0.189**

Visualization of the attention maps for CUB-200-2011 and Stanford Cars datasets are shown in Figure 7. Compared to ABN attention maps, MSABN maps are able to delineate the object boundaries better and provide information about the discriminative regions in the image. For CUB-200-2011 dataset, most of the attention in MSABN and MSABN+Puzzle is focused on the key attributes like the bill, wings, or the legs of the bird. The puzzle module performs significantly better than the rest of the model configurations here. This can be attributed to the effective regularization puzzle module provides in case of small datasets and the complex images (birds with different angles and actions). For Stanford Cars, the attention is focused on roughly similar regions depending on the car pose and form with different configurations. The performance improvement of MSABN and MSABN+Puzzle is not as significant here as for CUB-200-2011. This can be attributed to the nature of objects, i.e, cars, which have a well defined shape as compared to the birds which will have different shapes depending on the action like, flying, swimming, sitting or standing.

4.3   Attention Editing Performance

Datasets: We used the previously trained MSABN models on the CUB-200-2011 and Stanford Cars datasets to demonstrate the attention editing process. In both the datasets, bounding boxes of the objects within the image are available and are used here to mimic user input in the HITL pipeline in controlled experiments.

Experimental setup: We measured the accuracy with respect to the number of annotated samples (bounding boxes) "provided by the user" to estimate the human effort required to achieve performance gain. To decide which samples to "annotate", the attention maps of the training data were saved and a binary map was created by thresholding the intensity values at 0.2. Next, the ratio of attention inside and outside the bounding box of the total attention in the
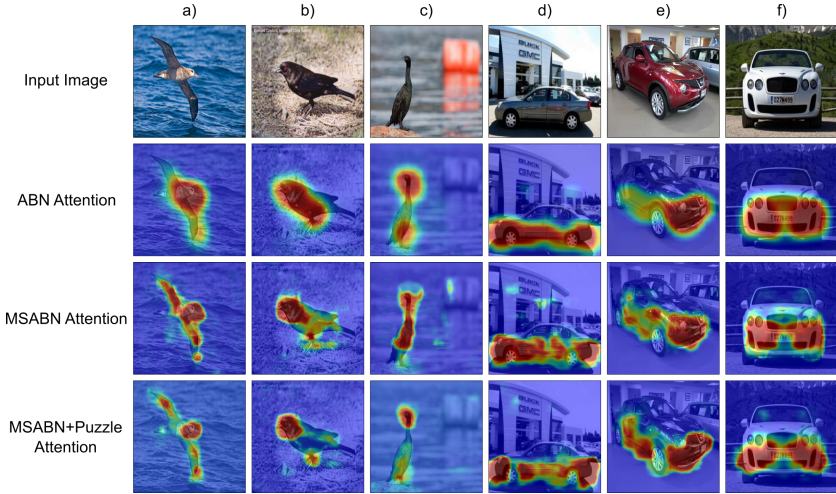
Fig. 7. Visualization of attention maps on the CUB-200-2011 (a, b, and c) and Stanford Cars (c, d, and e) dataset.

binary image were calculated. The number of samples for which the fraction of attention outside $frac_{attn\_out}$ was above a threshold λ_{out} were put in the bin of copy-replace augmentation for the fine-tuning training step. For the CUB-200-2011 dataset, the performance at $\lambda_{out} \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7\}$ was obtained and for Stanford Cars, the performance was obtained at $\lambda_{out} \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5\}$. We stopped at 0.7 and 0.5 respectively because the number of training samples were too low (less than 30) in some configurations to observe meaningful changes above these thresholds. We chose this way instead of measuring performance at different fractions of total number of training samples to better infer the trends of attention localization in the datasets and models. If the majority of the attention is focused inside the object box, the number of samples where $frac_{attn\_out} > \lambda_{out}$ would be much lower for higher thresholds. This method puts the images with worst attention localization first which of course "boosts" the reported results. However, a case can be made that this mimics a potential real scenario where a user corrects or edits discovered errors for fine-tuning a model: The training samples to show the user can be sorted either on the basis of wrong predictions first or the amount of overall attention (since the models might focus on background) to be shown first, which will make user input a little less time-consuming. The results were compared with results from vanilla fine-tuning with all the training images to better compare the effects of the copy-replace augmentation.

Training details: We evaluated all the MSABN models used in the fine-grained recognition. The models were optimized with SGD with a momentum of 0.9 and weight decay of 1e-4 for 50 epochs with a batch size of 16. The initial learning

rate was set to 0.1 and was divided by 10 at 50% and 75% of the total number of epochs. No augmentations were applied in the retraining to document the effects of only the copy-replace augmentation. The experiments were repeated three times for each model's (Resnet, ResNext, and EfficientNet) three repetitions in the previous section.
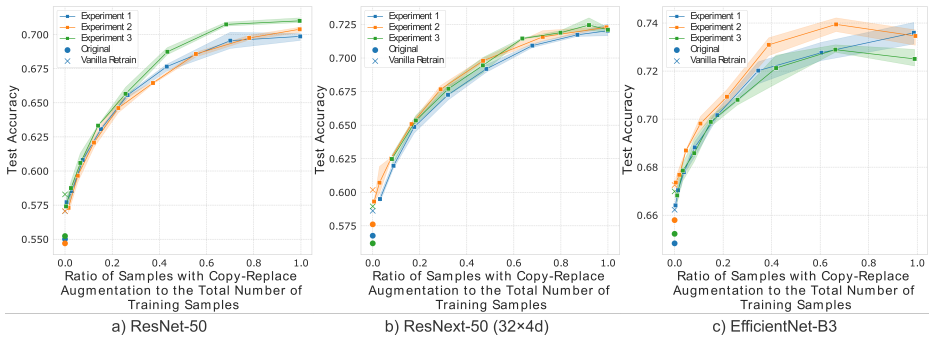

Fig. 8. Fine-tuning performance of different MSABN models on the CUB-200-2011 dataset with different ratios of training samples used for copy-replace augmentation.

Figure 8 and 9 show the results of the fine-tuning with copy-replace augmentation in CUB-200-2011 and Stanford Cars datasets. As can be seen from the figures, the fine-tuning improves performance of the models. The accuracy gain compared to vanilla retraining with only 20% of the training data augmented was approx. 6%, 5%, and 3% for the MSABN models of ResNet-50, ResNext-50, and EfficientNet-B3 respectively for the CUB-200-2011 dataset. For Stanford Cars, the accuracy increased by 2%, 1.5% and 0.5% respectively for the same amount of augmented training samples. Overall gain from augmenting almost all training samples compared with vanilla retraining was approx. 12.5%, 12.5%, and 6% for MSABN models of ResNet-50, ResNext-50, and EfficientNet-B3, respectively for the CUB-200-2011 dataset. For the Stanford dataset, the overall gain was approx. 4.5%, 3.5%, and 1.5%, respectively.

The distribution of data points along the x-axis in Figure 8 is relatively uniform compared to Figure 9 where most of the datapoints are located between zero and 0.2. This means that the number of training samples with $frac_{attn\_out} > \lambda_{out}$ changes proportionally with λ_{out} for the CUB-200-2011 dataset, but doesn't change much for the Stanford Cars dataset, which signals that the attention is more focused on the objects in the Stanford Cars dataset than in the CUB-200-2011 dataset. This can be attributed to the nature of objects in the datasets as mentioned earlier.

The experiments showed that a significant performance gain can be achieved with limited and relatively crude human input. As mentioned earlier, the results are biased towards the worst examples, so there is a need to develop a smart sorting system which decides the order of images to be shown to the user.
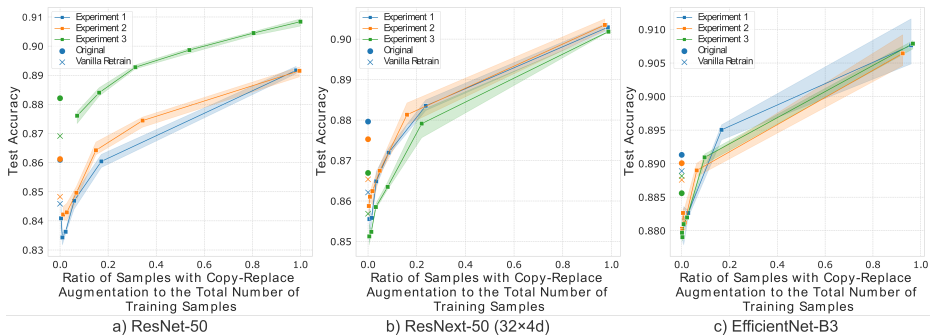
Fig. 9. Fine-tuning performance of different MSABN models on the Stanford Cars dataset with different ratios of training samples used for copy-replace augmentation.

We also noticed an interesting behavior with vanilla retraining on the two datasets. While the accuracy after retraining in CUB-200-2011 increased, the accuracy decreased with Stanford Cars dataset for all the models. The training loss curves (not shown) indicate that both models were destabilized (training loss increases) at the beginning of the training, however, in CUB-200-2011 dataset the models converged to a minima with lower validation loss than that of the initial training of the models. The observed effect in Stanford Cars dataset was the opposite. Further experiments to determine the cause were not done, instead, the performance was compared to the vanilla retrain as the benchmark.

Considerations: It's worth noting that due to upscaling of the outputs of intermediate blocks, the attention branch now has to process 16x more input values. This increases the computational cost of the MSABN models. We noticed that it takes roughly twice the time to train MSABN models compared to ABN models. The puzzle module increases the cost further as every image is processed twice, once as original and other as tiled sub-images during training, however, the model behaves like MSABN model during inference.

5   Conclusion

In this paper, we have presented a multi-scale attention branch network which greatly improves classification performance and also provide more accurate and detailed attention maps. We have evaluated the accuracy of MSABN for image recognition and fine-grained classification on multiple datasets and it was shown to outperform the ABN models. We also showed that using the puzzle module for fine-grained recognition increases the performance of the MSABN models. In addition, we introduced a HITL learning framework which inserts human knowledge in form of object bounding boxes and show that this is an effective way of improved the performance further.

References

1. Arai, S., Shirakawa, S., Nagao, T.: Non-strict attentional region annotation to improve image classification accuracy. In: 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC). pp. 2375–2380. IEEE (2021)
2. Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.R., Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one **10**(7), e0130140 (2015)
3. Branson, S., Perona, P., Belongie, S.: Strong supervision from weak annotation: Interactive training of deformable part models. In: 2011 International Conference on Computer Vision. pp. 1832–1839. IEEE (2011)
4. Branson, S., Wah, C., Schroff, F., Babenko, B., Welinder, P., Perona, P., Belongie, S.: Visual recognition with humans in the loop. In: European Conference on Computer Vision. pp. 438–451. Springer (2010)
5. Chattopadhay, A., Sarkar, A., Howlader, P., Balasubramanian, V.N.: Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In: 2018 IEEE winter conference on applications of computer vision (WACV). pp. 839–847. IEEE (2018)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
7. Dvornik, N., Mairal, J., Schmid, C.: Modeling visual context is key to augmenting object detection datasets. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 364–380 (2018)
8. Dwibedi, D., Misra, I., Hebert, M.: Cut, paste and learn: Surprisingly easy synthesis for instance detection. In: Proceedings of the IEEE international conference on computer vision. pp. 1301–1310 (2017)
9. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision **88**(2), 303–338 (2010)
10. Fukui, H., Hirakawa, T., Yamashita, T., Fujiyoshi, H.: Attention branch network: Learning of attention mechanism for visual explanation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10705–10714 (2019)
11. Ghiasi, G., Cui, Y., Srinivas, A., Qian, R., Lin, T.Y., Cubuk, E.D., Le, Q.V., Zoph, B.: Simple copy-paste is a strong data augmentation method for instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2918–2928 (2021)
12. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. pp. 1026–1034 (2015)
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
14. Jo, S., Yu, I.J.: Puzzle-cam: Improved localization via matching partial and full features. In: 2021 IEEE International Conference on Image Processing (ICIP). pp. 639–643. IEEE (2021)
15. KC, D., Zhang, C.: Improving the trustworthiness of image classification models by utilizing bounding-box annotations. arXiv preprint arXiv:2108.10131 (2021)

16. Kim, J.H., Choo, W., Song, H.O.: Puzzle mix: Exploiting saliency and local statistics for optimal mixup. In: International Conference on Machine Learning. pp. 5275–5285. PMLR (2020)

17. Koziarski, M., Cyganek, B., Olborski, B., Antosz, Z., Żydak, M., Kwolek, B., Wasowicz, P., Bukała, A., Swadźba, J., Sitkowski, P.: Diagset: a dataset for prostate cancer histopathological image classification. arXiv preprint arXiv:2105.04014 (2021)

18. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13). Sydney, Australia (2013)

19. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)

20. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25** (2012)

21. Linsley, D., Shiebler, D., Eberhardt, S., Serre, T.: Learning what and where to attend. In: International Conference on Learning Representations (2019)

22. Mitsuhara, M., Fukui, H., Sakashita, Y., Ogata, T., Hirakawa, T., Yamashita, T., Fujiyoshi, H.: Embedding human knowledge into deep neural network via attention map. arXiv preprint arXiv:1905.03540 (2019)

23. Parikh, D., Grauman, K.: Interactively building a discriminative vocabulary of nameable attributes. In: CVPR 2011. pp. 1681–1688. IEEE (2011)

24. Parkash, A., Parikh, D.: Attributes for classifier feedback. In: european conference on computer vision. pp. 354–368. Springer (2012)

25. Ribeiro, M.T., Singh, S., Guestrin, C.: " why should i trust you?" explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. pp. 1135–1144 (2016)

26. Rieger, L., Singh, C., Murdoch, W., Yu, B.: Interpretations are useful: penalizing explanations to align neural networks with prior knowledge. In: International conference on machine learning. pp. 8116–8126. PMLR (2020)

27. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Gradcam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017)

28. Smilkov, D., Thorat, N., Kim, B., Viégas, F., Wattenberg, M.: Smoothgrad: removing noise by adding noise. arXiv preprint arXiv:1706.03825 (2017)

29. Springenberg, J.T., Dosovitskiy, A., Brox, T., Riedmiller, M.: Striving for simplicity: The all convolutional net. arXiv preprint arXiv:1412.6806 (2014)

30. Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: International conference on machine learning. pp. 6105–6114. PMLR (2019)

31. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD Birds-200-2011 Dataset. Tech. Rep. CNS-TR-2011-001, California Institute of Technology (2011)

32. Wang, H., Gong, S., Zhu, X., Xiang, T.: Human-in-the-loop person re-identification. In: European conference on computer vision. pp. 405–422. Springer (2016)

33. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1492–1500 (2017)

34. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6023–6032 (2019)
35. Zhang, Q.s., Zhu, S.C.: Visual interpretability for deep learning: a survey. Frontiers of Information Technology & Electronic Engineering **19**(1), 27–39 (2018)
36. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2921–2929 (2016)