EDDA: Explanation-driven Data Augmentation to Improve Model and Explanation Alignment

Ruiwen Li [†]
University of Toronto
ruiwen.li@mail.utoronto.ca

Zhibo Zhang [†]
University of Toronto
zhibozhang@cs.toronto.edu

Jiani Li
University of Toronto
nini.li@mail.utoronto.ca

Scott Sanner
University of Toronto
ssanner@mie.utoronto.ca

Jongseong Jang
LG AI Research
j.jang@lgresearch.ai

Yeonjeong Jeong
LG AI Research
yj.jeong@lgresearch.ai

Dongsub Shim
LG AI Research
dongsub.shim@lgresearch.ai

Abstract

Recent years have seen the introduction of a range of methods for post-hoc explainability of image classifier predictions. However, these post-hoc explanations may not always align perfectly with classifier predictions, which poses a significant challenge when attempting to debug models based on such explanations. To this end, we seek a methodology that can improve alignment between model predictions and explanation method that is both agnostic to the model and explanation classes and which does not require ground truth explanations. We achieve this through a novel explanation-driven data augmentation (EDDA) method that augments the training data with occlusions of existing data stemming from model-explanations; this is based on the simple motivating principle that occluding salient regions for the model prediction should decrease the model confidence in the prediction, while occluding non-salient regions should not change the prediction — if the model and explainer are aligned. To verify that this augmentation method improves model and explainer alignment, we evaluate the methodology on a variety of datasets, image classification models, and explanation methods. We verify in all cases that our explanation-driven data augmentation method improves alignment of the model and explanation in comparison to no data augmentation and non-explanation driven data augmentation methods. In conclusion, this approach provides a novel model- and explainer-agnostic methodology for improving alignment between model predictions and explanations, which we see as a critical step forward for practical deployment and debugging of image classification models.

1 Introduction

Deep learning has become the mainstream methodology since it was applied to image classification tasks [1]. However, most deep learning models are black boxes. Employing these models in high-risk domains such as healthcare [2] and stock market prediction [3] could cause a high price because of a minor mistake. Therefore, model understanding is imperative in such domains, and explaining deep models has been drawing the attention of the machine learning community [2, 4, 5, 3, 6–10, 4, 11–15].

[†] Equal contribution, co-first authors.

Preprint. Under review.

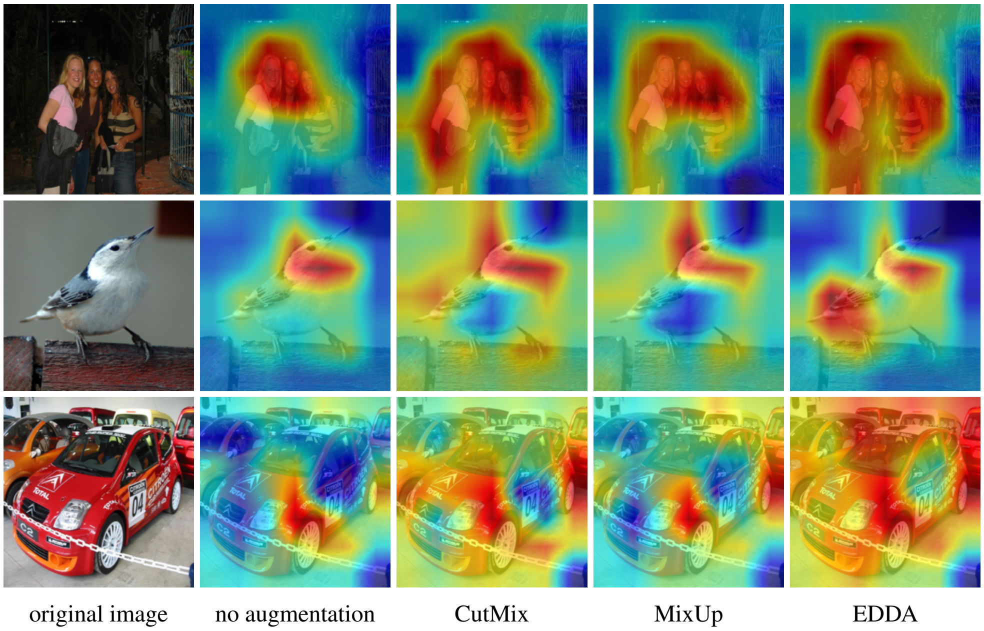

Figure 1: Some example heatmap visualizations. As seen in the graphs, our method (EDDA) shows more complete consideration and focus on the entire object when making decisions. This is crucial in high-risk domains, since partially focus on an object in decision-making can indicate overfitting and bias, which can easily cause mistakes on unseen data.

In recent years, post-hoc explanations are primarily used in explaining pre-built black-box models and facilitating model debugging. However, there is always a concern about whether we should trust the explanations given by some post-hoc explanation methods when we use them to debug the deep learning models. According to our knowledge, there is no existing literature that focuses on utilizing existing explanation methodologies to improve the alignment between model prediction and explanations. An approach to realize this goal in an explainer- and model-agnostic manner is data augmentation. Existing works like CutMix [16] and MixUp [17] augment the data to increase the generalization capability of the deep models, but neither ever aimed to improve the explanation faithfulness. In this work, we introduce our explainer- and model-agnostic explanation-driven data augmentation technique - EDDA. To motivate our work enough, first of all, what will happen if the explanations by Deep Neural Nets do not reliably align with their predictions?

To answer this question, we show some examples in Figure 1. The left column contains several images from the PASCAL VOC dataset [18]. The successive four columns are GradCAM [19] visualizations from the VGG-16 models based on the vanilla training process, training with CutMix [16] data augmentation, training with MixUp [17] data augmentation, as well as training with our proposed augmentation method - EDDA. Among these heatmap visualizations, the red parts represent the positive contributions towards the prediction, and the blue parts represent the negative contributions towards the prediction, and the color in between has a smaller magnitude of contribution towards the predictions - whether positive or negative. When we use deep learning models to classify a car object, as shown in the third row of Figure 1, the models trained without data augmentation ('no augmentation' column in the figure), CutMix augmentation as well as MixUp augmentation all focus on the partial object (mainly the head and the door of the car) to make predictions. Ironically, parts of the car also contribute negatively towards their predictions (the doors of the car for all three intermediate columns).

An extension to real-world application would be - suppose in a self-driving car scenario where we use computer vision to detect pedestrians. The explanation of the classifier tells you that: 1. the system is relying on parts of the human body rather than the entire object to make decisions; 2. some parts of the human body are making negative contributions towards the classifier's prediction as a pedestrian. Knowing this, do you still dare to walk on the road with self-driving cars passing beside you? The answer is surely no. Since both cases indicate overfitting and classification bias and increase mis-classification probability on the unseen data.

Examples above verify the fact that having explanations alone is not enough, knowing how to improve and debug machine learning systems such that they are more reliable are critical. Motivated by these concerns, we propose our novel method EDDA - Explanation-driven Data Augmentation, which augments the data by masking out the regions of high importance based on the feature attribution methods if the explainer gives unfaithful explanations. We feed the perturbed images into the classifier during the training process and force the classifier to focus on other input regions to make decisions. Moreover, this augmentation can increase the chance that the classifier makes decisions based on more appropriate reasons.

To be specific, our contributions in this paper are as follows:

1. We propose our novel explanation-driven data augmentation (EDDA) methods that utilize existing post-hoc explanation methodologies for both multi-class classification and multi-label classification settings;
2. We propose a training framework that produces and employs the augmented data at the same time along the training process;
3. We conduct extensive experiments on the CIFAR-100 [20], PASCAL VOC 2012 [18] and Oxford-IIIT Pet [21] datasets, evaluating through the Grad-CAM [19] and Saliency Map [11] explanation methods. We empirically show that our method gives explanations that are more loyal to the predictor across datasets, deep learning models, and explanation methods, both visually and quantitatively.

As shown in the rightmost column of Figure 1, our method can better focus on the body part of the classification object and make more confident predictions towards the target class.

2 Related Work

Post-hoc Explanations Post-hoc methods explain the decisions made by a pre-built black-box model. Most of the current post-hoc explanation methods focus on local explanations, which describe the model behavior at the neighborhood of an individual prediction. Early post-hoc explanation methods, such as LIME [10] and SHAP [9], attribute importance scores to input features with respect to the prediction of the target data point. Gradient-based methods are broadly used in explaining Convolutional Neural Networks (CNNs): Saliency Map [11] utilizes the input gradient of the network to highlight the critical pixels for a prediction; Integrated Gradients [8] addresses the gradient saturation issue by generating interpolants from the baseline to the input image and summing over the gradient of each interpolant; SmoothGrad [7] addresses the same issue, and it computes sharper saliency maps by adding noise to copies of the input image and averaging the input gradient of these copies. The Class Activation Map (CAM) [22] based explanation methods are intensively researched. Grad-CAM [19] employs the weighted activations of the feature maps in the final convolutional layer, where the weights are computed based on the gradient of an output class with respect to each feature map. Despite the variety of these methods, they were either evaluated qualitatively or based on ground truth localization information such as object detection bounding boxes. It may not be meaningful if the dataset is biased, e.g., the classifier can utilize the sea background of the fish objects or the sky background of the plane object to make decisions that overfits specific training data. Recent image-based explanation methods such as Grad-CAM++ [4], RISE [23], Score-CAM [24], and SISE [25] considered the metrics of faithfulness, which evaluates the alignment between the explanation and the model prediction. However, they did not come up with the idea of how to increase explanation faithfulness.

Data Augmentation Data augmentation is a training strategy that improves the performance of deep learning models by applying various transformations to the original training data. These transformations are relatively simple but effective in increasing the data diversity and the model's generalization ability. Cutout [12] is a regional dropout method that zero-masks a random fix-sized region of each input training image to improve the model's test accuracy. MixUp [17] linearly combines two training inputs where their targets are linearly interpolated in the same fashion. CutMix [16] builds on Cutout, and it avoids the information loss in the dropout region by randomly removing

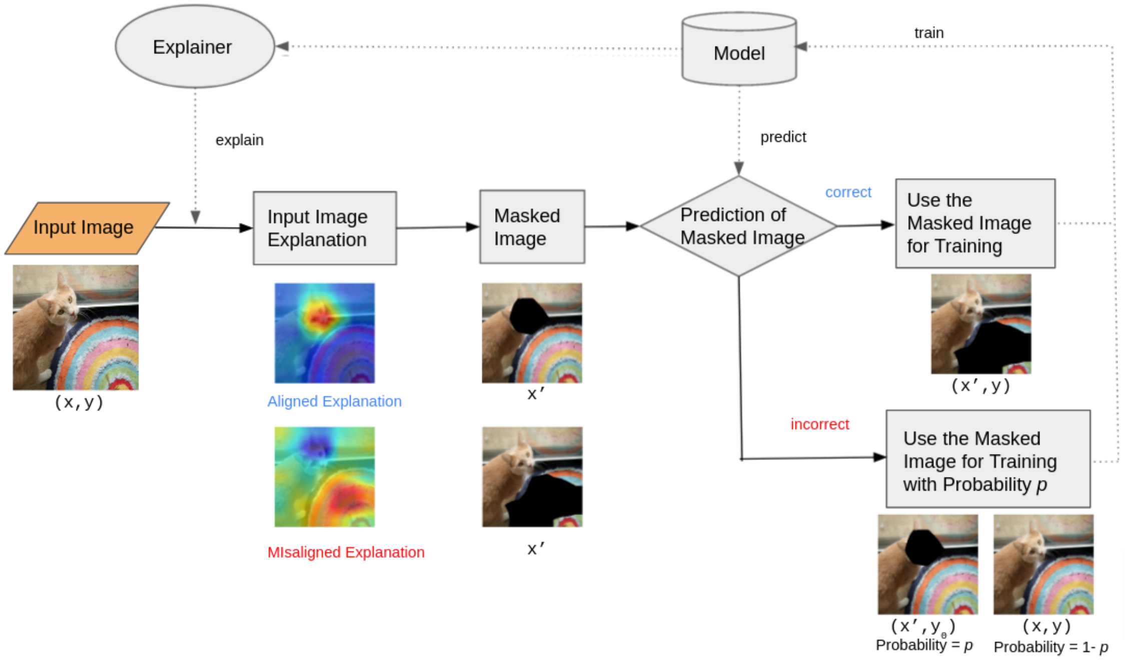

Figure 2: Explainer-driven data augmentation for multi-class image classification

and mixing patches among training images where the target labels assigned are proportional to the size of those patches. These methods show improved test accuracy, but they have never been verified from the perspective of explanation. In this paper, we provide a novel augmentation method that improves explanation faithfulness and the verification of existing data augmentation methods in terms of explanation faithfulness. This verification is critical for identifying whether the accuracy improvement is benefited from the bias in the dataset.

3 Proposed Method

In this section, we describe the proposed explainer-driven data augmentation (EDDA) method in detail. Our method employs an explanation method that attributes predictions to pixels or contiguous regions of the input image. Most of the existing saliency methods apply here. However, backpropagation-based approaches without sampling such as [11, 19, 26] are preferred in consideration of training speed.

3.1 Motivation

Suppose we have a classifier, and we explain the model prediction of an input image. If we have access to the ground truth localization information of the predicted object, we can easily measure whether the explanation is aligned with the prediction by comparing it with the localization information. If they do not align, we can mask out the image parts that the classifier incorrectly focused on and train on the masked image so that the model will focus on some other region for the prediction. However, it is costly to label object localization in real-world applications. We can still assess the alignment by masking out the region that the model is focusing on. If the prediction score significantly drops, it suggests that the explanation is well aligned with the prediction, and we adopt the original image for training. Otherwise, we train on the masked image to induce the model to focus on other regions. Motivated by this idea, we propose our explainer-driven data augmentation method.

3.2 EDDA for Multi-class Classification - EDDA$_{mc}$

Figure 2 shows the training process of our method in the multi-class classification setting. Let $x \in \mathbb{R}^{W \times H \times C}$ and $x \in [0, 1]$ denote an original image with a ground truth label y from the training set, and let $f_{\mathcal{M}}(\cdot)$ denote the explainer employed to explain the model $\mathcal{M}$ which is being trained. We use the explainer to obtain the saliency map $x_{saliency}$ of the input image with respect to y, which indicates the attribution of pixels to the prediction score of the target label. This process can be formally defined as

$$x^{y}_{saliency} = f_{\mathcal{M}}(x, y) \tag{1}$$

where $x^{y}_{saliency} \in \mathbb{R}^{W \times H}$ and it is normalized into the range of $[0, 1]$. Then, we find the most salient pixels that are above some pre-defined threshold τ and we occlude these pixels in the original image as

$$x_{mask} = x \odot (x^{y}_{saliency} > \tau) \tag{2}$$

where $x_{mask} \in \mathbb{R}^{W \times H \times C}$ denotes the perturbed image. We obtain the prediction $\hat{y}$ on the perturbed image x_{mask} by feeding it into the classifier $\mathcal{M}$.

If the prediction is correct after masking the 'important' region, i.e $\hat{y} = y$, it indicates that the salient pixels do not help make the correct prediction, thus the explanation is likely not well aligned with the model prediction. In this case, we add the masked image x_{mask} with the original label y into the training set. In this way, we force the classifier to focus on other regions of the input for classification. Otherwise, we either adopt the masked image x_{mask} associated with a background label y_0 with probability p (a hyperparameter) or use the original image for training. The full algorithm is in algorithm 2.

Algorithm 1: Training Pipeline for Explanation-driven Data Augmentation

```
Input: train set $\mathcal{D}$, classifier $\mathcal{M}$, explainer $f_{\mathcal{M}}(\cdot)$, training epochs $E$, mini-batch size $N$
for $e \in \{1, ..., E\}$ do
    for $(\mathcal{X}, \mathcal{Y}) \overset{N}{\sim} \mathcal{D}$ do
        $\tilde{\mathcal{X}}, \tilde{\mathcal{Y}} \leftarrow$ EDDA$(\mathcal{X}, \mathcal{Y}, \mathcal{M}, f_{\mathcal{M}})$    ▷ use Algorithm 2 or Algorithm 3 depending on the task
        $\mathcal{M} \leftarrow$ Train$(\mathcal{M}, \tilde{\mathcal{X}}, \tilde{\mathcal{Y}})$
    end
end
```

3.3 EDDA for Multi-label Classification - EDDA$_{ml}$

We propose a separate method in the multi-label classification setting, where there are multiple ground truth labels for a single image. A multi-label classifier can be considered as a combination of multiple binary classifiers. This raises the needs for doing independent explanations for different labels. We first make predictions on the input image, then select the set of classes that have positive predictions and align with the ground truth label - the `True-Positive` classes. We explain the input image with respect to those classes and acquire the according masked images. For each masked image, if the prediction is correct on the according label, then we add that masked image and label into the training batch. The full algorithm is suggested in 3.

Algorithm 2: EDDA$_{mc}$: Explanation-driven Data Augmentation for Multi-class Classification

```
Input: input image batch $\mathcal{X}$, target label batch $\mathcal{Y}$, model $\mathcal{M}$, explainer $f_{\mathcal{M}}(\cdot)$
$\mathcal{X}', \mathcal{Y}' \leftarrow \{\}, \{\}$
// Get the attribution map using explainer $f_{\mathcal{M}}$
$\mathcal{X}_{saliency} \leftarrow f_{\mathcal{M}}(\mathcal{X}, \mathcal{Y})$                              ▷ Equation (1)
// Mask the most salient regions using a pre-defined hyperparameter $\tau$
$\mathcal{X}_{mask} \leftarrow \mathcal{X} \odot (\mathcal{X}_{saliency} > \tau)$                        ▷ Equation (2)
// Predict the labels based on masked images.
$\hat{\mathcal{Y}}_{mask} \leftarrow \mathcal{M}(\mathcal{X}_{mask})$
// Predict the labels based on original images.
$\hat{\mathcal{Y}} \leftarrow \mathcal{M}(\mathcal{X})$
for $1 \leq k \leq |\mathcal{X}_{mask}|$ do
    if $\mathcal{Y}[k] = \hat{\mathcal{Y}}_{mask}[k]$ then
        // If the prediction on the masked image aligns with the ground truth label, then adopt the
        // masked image and the original label for training.
        $\mathcal{X}' \leftarrow \mathcal{X}' \cup \{\mathcal{X}_{mask}[k]\}$
        $\mathcal{Y}' \leftarrow \mathcal{Y}' \cup \{\mathcal{Y}[k]\}$
    else
        $r \sim \text{Uniform}(0, 1)$
        if $r > p$ then
            // Adopt the masked image and the background label for training.
            $\mathcal{X}' \leftarrow \mathcal{X}' \cup \{\mathcal{X}_{mask}[k]\}$
            $\mathcal{Y}' \leftarrow \mathcal{Y}' \cup \{background\}$
        else
            // Adopt the original image and the original label for training.
            $\mathcal{X}' \leftarrow \mathcal{X}' \cup \{\mathcal{X}[k]\}$
            $\mathcal{Y}' \leftarrow \mathcal{Y}' \cup \{\mathcal{Y}[k]\}$
end
return $\mathcal{X}', \mathcal{Y}'$
```

Algorithm 3: EDDA$_{ml}$: Explanation-driven Data Augmentation for Multi-label Classification

```
Input: input image batch $\mathcal{X}$, target label batch $\mathcal{Y}$, model $\mathcal{M}$, explainer $f_{\mathcal{M}}(\cdot)$
$\mathcal{X}', \mathcal{Y}' \leftarrow \mathcal{X}, \mathcal{Y}$
for $1 \leq k \leq |\mathcal{X}|$ do
    // Obtain the prediction given by the classifier.
    $\hat{\mathcal{Y}} \leftarrow \mathcal{M}(\mathcal{X}[k])$
    // Comparing with the ground truth labels, we select the True-Positive classes (where the
    // prediction and the label are both True) and denote the set to be $c$
    $\mathcal{C} \leftarrow indices(\hat{\mathcal{Y}} = \mathcal{Y}[k])$
    // Get the importance maps for all correct labels at the same time using explainer $f_{\mathcal{M}}$.
    $\mathcal{X}_{saliency} \leftarrow f_{\mathcal{M}}(\mathcal{X}[k], \mathcal{C})$
    // Mask the most salient regions using a pre-defined hyperparameter $\tau$.
    $\mathcal{X}_{mask} \leftarrow \mathcal{X}[k] \odot (\mathcal{X}_{saliency} > \tau)$
    // Predict the labels based on masked images.
    $\hat{\mathcal{Y}}_{mask} \leftarrow \mathcal{M}(\mathcal{X}_{mask})$
    for $1 \leq z \leq |\mathcal{C}|$ do
        if $\hat{\mathcal{Y}}_{mask}[z] = \mathcal{Y}[k, z]$ then
            $\mathcal{X}' \leftarrow \mathcal{X}' \cup \{\mathcal{X}_{mask}[z]\}$
            $\mathcal{Y}' \leftarrow \mathcal{Y}' \cup \{\mathcal{Y}[k, z]\}$
        end
    end
end
return $\mathcal{X}', \mathcal{Y}'$
```

4 Experiments

This section compares our method with no data augmentation and state-of-the-art non-explainer-driven data augmentation methods including CutMix [16] and MixUp [17]. We verify our method's performance on popular convolution networks, including VGG-16 [27] and ResNet-50 [28]. We conduct experiments on the CIFAR-100 [20], PASCAL VOC 2012 [18] as well as Oxford-IIIT Pet [21] datasets.

4.1 Evaluation Metrics

In order to find a metric that truly evaluates the faithfulness of the explanations, our motivation is: can we perturb the inputs, propagate from the inputs to the decisions and then observe the changes in the decisions? In this way, it is ensured that any change in the output probabilities is directly induced by the explanation-guided perturbations. We find that the Drop% and the Increase% introduced in [29, 30, 25] exactly match our criteria. Drop% measures the positive attribution loss after masking out unimportant regions, while Increase% measures the negative attribution discard [25]. The masked image should preserve the positive attribution and discard the negative attribution if the explanation is well aligned with the prediction. These metrics do not require ground truth bounding boxes for the classified objects, which are expensive to label especially in real-world applications. In fact, bounding boxes may not be a good ground truth for our purpose since they contain extra pixels irrelevant to the object, and they are not labeled from the model perspective. We extend these two metrics, where for both Drop% and Increase% we measure

1. the *proportion* of examples whose confidence score with respect to the predicted label drops/increases after occluding high salient regions, which is defined as

$$Drop\%_{prop} = \frac{1}{N}\sum_{i=1}^{N} \text{sign}(\mathcal{M}^c(x_i \odot T(x_{i_{saliency}})) - \mathcal{M}^c(\mathcal{X})) \times 100$$

$$Increase\%_{prop} = \frac{1}{N}\sum_{i=1}^{N} \text{sign}(\mathcal{M}^c(x_i) - \mathcal{M}^c(x_i \odot T(x_{i_{saliency}}))) \times 100$$

   where x_i represents the input image, $\mathcal{M}$ represents the classifier, c represents the predicted label, $T(\cdot)$ represents a threshold function that is applied on the saliency map $x_{i_{saliency}}$ to extract the top salient pixels.

2. the average *magnitude* of drop and increase of the confidence with respect to the predicted label after occluding high salient regions, which is defined as

$$Drop\%_{mag} = \frac{1}{N}\sum_{i=1}^{N} \frac{\max(0, \mathcal{M}^c(x_i \odot T(x_{i_{saliency}})) - \mathcal{M}^c(x_i))}{\mathcal{M}^c(x_i)} \times 100$$

$$Increase\%_{mag} = \frac{1}{N}\sum_{i=1}^{N} \frac{\max(0, \mathcal{M}^c(x_i) - \mathcal{M}^c(x_i \odot T(x_{i_{saliency}})))}{\mathcal{M}^c(x_i)} \times 100$$

In our experiments, we used the top 15% as the threshold (following previous work [25]) for selecting the most salient pixels for function $T(\cdot)$. Explanations better aligned with the predictions are expected to have lower Drop% and higher Increase%, since good explanations will highlight the most critical image regions for making the prediction [29].

4.2 Comparison Methods

No Augmentation Since our augmentation method is based on existing explainers like [11, 19], one natural question that we want to ask is: How well will these explainers perform without the assistance of our data augmentations? In order to examine this, for every explanation method that our augmentation was based on in the experiment, we evaluated its explanation performance on the test set based on the model trained without augmentation to compare the model trained with our EDDA method.

CutMix, MixUp In order to thoroughly verify the explanation capability of our EDDA method, another question we want to answer empirically is: Will our method be superior compared with other SOTA augmentation methods? CutMix [16] and MixUp [17] are two influential augmentation methods that achieve SOTA results in test accuracy across many popular datasets. CutMix belongs to a type of regional dropout method. It randomly removes some image regions and fills them with patches from other training images. The target labels are assigned based on the proportion of the area of those patches. In contrast, MixUp does not use regional dropout. It linearly blends two training inputs. The targets of the augmented images are assigned based on the blending ratio of their source images.

4.3 Quantitative Evaluation Approaches

For each of the datasets used in our experiment, we trained the VGG-16 and ResNet-50 models with weights initialized from models pre-trained on ImageNet [31]. We trained models with different data augmentation strategies, including CutMix, MixUp, and our proposed explainer-driven data augmentation method. As a comparison, we also trained models without data augmentation. We based our method on Grad-CAM and Saliency Map explainers for the training process. The hyperparameter p was set to 0 in all experiments, i.e., we used the original image for training if the prediction on the masked image was incorrect. The saliency threshold τ was set to 0.5, and the learning rate, momentum, weight decay were set to 0.01, 0.9, and 0.0001, which we found work well across all experimented datasets.

We experimented our multi-label classification augmentation method on the PASCAL VOC 2012 [18] dataset. We experimented our multi-class classification augmentation method on the CIFAR-100 [20] and Oxford-IIIT Pet [21] datasets. Experiments were conducted on the CIFAR-100 and PASCAL VOC datasets on their test sets. For the Oxford-IIIT Pet dataset, we randomly split the data and use 75% of the data for training and the rest for testing. For the experiments on CIFAR-100, we trained each model for 200 epochs, and we used standard augmenatation such as cropping and flipping following the setting in [17, 16]. For experiments on the PASCAL VOC 2012 and Oxford-IIIT Pet datasets, we trained each model for 20 epochs, and the input images were resize to 224×224 without other standard augmentation. We report the performance of different methods in terms of Drop% and Increase% as introduced in 4.1. The numbers reported are based on the average over three independent runs with different random seeds. The results on the three datasets about the performance in Drop%$_{\text{prop}}$ and Increase%$_{\text{prop}}$ are shown in Table 1 above. We put the results about the performance in Drop%$_{\text{mag}}$ and Increase%$_{\text{mag}}$ in Table 1 in supplementary materials, due to space limitation.

4.4 Explanation Visualizations

We examine some example explanations across the models trained under different augmentation methods as well as no augmentation method. We visualized them using heat maps, as shown in Figure 1. We also included some more visualizations in the supplementary materials. For the detailed discussion of the visualizations, please refer to Section 1.

4.5 Discussion

According to the experimental results, for both cases (whether under degree changes or not), our method shows dramatic improvement in both metrics - a larger increase score and a smaller drop score, while CutMix and MixUp do not show persuasive advantages over the models trained without augmentation. The larger scores of both, as discussed earlier in 4.1, indicate that the explanations based on our method are more aligned with the classifier's predictions, since the more important the input region is based on the explanations, the more it will affect the network's decisions. This result suggests our method helps achieve our initial goal of this research - better alignment between explanations and model predictions.

Besides, the advantageous performance of our method is consistent across datasets, explainers and deep learning models, which further verifies that fact that our explanation-driven data augmentation method is model- and explainer-agnostic and have the potential to provide faithful explanations in all cases. We observe that EDDA achieved the best accuracy among comparison methods using VGG16 model on the PASCAL VOC dataset. We also observe that the model trained with EDDA

Model	ResNet-50				VGG-16			
Metric	Drop		Increase		Drop		Increase	
f	GC	SA	GC	SA	GC	SA	GC	SA
No Aug	99.24	99.49	0.76	0.51	98.82	99.37	1.01	0.58
CutMix	99.64	99.69	0.36	0.31	99.49	99.80	0.51	0.20
MixUp	99.18	99.45	0.82	0.55	98.68	99.47	1.32	0.53
EDDA$_{\text{GC}}$	**95.65**	99.08	**4.28**	0.92	**95.68**	98.54	**3.53**	1.24
EDDA$_{\text{SA}}$	99.22	**98.57**	0.78	**1.42**	99.07	**97.58**	0.87	**1.71**

(a) CIFAR-100

Model	ResNet-50				VGG-16			
Metric	Drop		Increase		Drop		Increase	
f	GC	SA	GC	SA	GC	SA	GC	SA
No Aug	92.02	99.74	7.96	0.26	87.82	99.45	10.25	0.53
CutMix	94.53	99.69	5.47	0.31	93.02	99.56	6.98	0.44
MixUp	80.83	**95.94**	19.17	**4.06**	89.39	99.27	10.61	0.73
EDDA$_{\text{GC}}$	**79.47**	98.95	**20.33**	1.05	80.13	99.36	**19.62**	0.63
EDDA$_{\text{SA}}$	92.65	98.48	7.24	1.52	**79.58**	**97.34**	11.39	**2.40**

(b) PASCAL VOC 2012

Model	ResNet-50				VGG-16			
Metric	Drop		Increase		Drop		Increase	
f	GC	SA	GC	SA	GC	SA	GC	SA
No Aug	96.53	99.96	3.47	0.04	93.06	99.53	5.82	0.40
CutMix	92.99	99.93	7.01	0.07	96.49	99.96	3.51	0.04
MixUp	91.03	99.93	8.97	0.07	91.86	99.93	8.14	0.07
EDDA$_{\text{GC}}$	**85.79**	99.93	**14.21**	0.07	**78.52**	99.71	**16.49**	0.25
EDDA$_{\text{SA}}$	95.77	**99.78**	4.23	**0.22**	92.55	**98.34**	6.62	**1.48**

(c) Oxford-IIIT Pet Dataset

Table 1: ResNet-50 and VGG-16 model performance in terms of Drop%$_{\text{prop}}$ and Increase%$_{\text{prop}}$ on CIFAR-100, PASACL VOC 2012, and Oxford-IIIT Pet Dataset. We report the performance of both employing Grad-CAM (GC) and Saliency Map (SA) as the saliency method f used on the threshold function $T(\cdot)$ when calculating Drop% and Increase%. For our method, we report the performance of the models that trained using GC and SA as the explainers. We compare the performance of our method with CutMix, MixUp, as well as no augmentation (No Aug).

has 1-2% drop in test accuracy compared to No Aug on CIFAR-100, which is possibly the result of test dataset bias. We do believe that the models with more faithful explanations will eventually have better generalization capability on unseen data, if the testing set is broad enough to truly cover the underlying data distributions.

5 Conclusions and Future Work

In this work, we propose a novel explanation-driven data augmentation method that can employ existing post-hoc explanation methods to improve the explanation faithfulness of black box models. We evaluate our method empirically on different datasets, explainers as well as deep learning models and verify that our method is both explainer-agnostic and model-agnostic. Through quantitative and visualization examination, our method shows advantages compared to: 1. models trained without data augmentations; 2. models trained with state-of-the-art data augmentation methods. Through the better Drop% and Increase% scores, it is obvious that the models trained with our method gives explanations whose important regions affect the classifier decisions to a larger extent, whose unimportant regions affect the decision making to a smaller degree, compared to other comparison approaches. We conclude that the models trained with EDDA improve the alignment between explanations and predictions.

However, our augmentation method is currently only verified in the input space for image classification tasks. As part of the future work, we plan to: 1.generalize our method to latent variable models and check the performance of EDDA in the embedding space; 2. generalize EDDA to the deep time series domain. We also hope that our method can serve as a starting point and raise more interest and attention in model debugging using explanations in the XAI community.

References

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012.

[2] Andre Esteva, Alexandre Robicquet, Bharath Ramsundar, Volodymyr Kuleshov, Mark DePristo, Katherine Chou, Claire Cui, Greg Corrado, Sebastian Thrun, and Jeff Dean. A guide to deep learning in healthcare. *Nature medicine*, 25(1):24–29, 2019.

[3] Eunsuk Chong, Chulwoo Han, and Frank C Park. Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Systems with Applications*, 83:187–205, 2017.

[4] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847. IEEE, 2018.

[5] Ian Covert, Scott Lundberg, and Su-In Lee. Understanding global feature contributions with additive importance measures. *Advances in Neural Information Processing Systems*, 33, 2020.

[6] Yash Goyal, Ziyan Wu, Jan Ernst, Dhruv Batra, Devi Parikh, and Stefan Lee. Counterfactual visual explanations. In *International Conference on Machine Learning*, pages 2376–2384. PMLR, 2019.

[7] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

[8] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR, 2017.

[9] Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *arXiv preprint arXiv:1705.07874*, 2017.

[10] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " 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*, pages 1135–1144, 2016.

[11] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[12] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[13] Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. *Advances in Neural Information Processing Systems*, 33, 2020.

[14] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning*, pages 1885–1894. PMLR, 2017.

[15] Chih-Kuan Yeh, Joon Sik Kim, Ian EH Yen, and Pradeep Ravikumar. Representer point selection for explaining deep neural networks. *arXiv preprint arXiv:1811.09720*, 2018.

[16] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019.

[17] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[18] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[19] 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 *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[21] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012.

[22] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[23] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*, 2018.

[24] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 24–25, 2020.

[25] Sam Sattarzadeh, Mahesh Sudhakar, Anthony Lem, Shervin Mehryar, KN Plataniotis, Jongseong Jang, Hyunwoo Kim, Yeonjeong Jeong, Sangmin Lee, and Kyunghoon Bae. Explaining convolutional neural networks through attribution-based input sampling and block-wise feature aggregation. *arXiv preprint arXiv:2010.00672*, 2020.

[26] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[27] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[29] Harish Guruprasad Ramaswamy et al. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 983–991, 2020.

[30] Ruigang Fu, Qingyong Hu, Xiaohu Dong, Yulan Guo, Yinghui Gao, and Biao Li. Axiom-based grad-cam: Towards accurate visualization and explanation of cnns. *arXiv preprint arXiv:2008.02312*, 2020.

[31] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

Supplementary Materials

Model	ResNet-50				VGG-16			
Metric	Drop		Increase		Drop		Increase	
f	GC	SA	GC	SA	GC	SA	GC	SA
No Aug	96.18	97.97	0.28	0.22	93.43	97.71	0.24	0.12
CutMix	96.50	98.30	0.15	0.22	95.34	98.10	0.20	0.08
MixUp	94.38	96.41	0.60	**0.63**	94.24	96.95	**0.89**	**0.46**
EDDA$_{\text{GC}}$	**86.09**	96.68	**0.97**	0.32	**87.80**	95.63	0.63	0.27
EDDA$_{\text{SA}}$	96.53	**95.33**	0.27	0.42	94.75	**92.65**	0.20	0.30

(a) CIFAR-100

Model	ResNet-50				VGG-16			
Metric	Drop		Increase		Drop		Increase	
f	GC	SA	GC	SA	GC	SA	GC	SA
No Aug	52.59	96.96	1.48	0.14	40.84	95.93	2.98	0.37
CutMix	60.45	93.82	1.92	0.31	48.93	91.22	1.75	0.16
MixUp	54.66	93.94	2.78	**1.55**	45.01	90.73	2.30	0.23
EDDA$_{\text{GC}}$	**37.89**	**81.96**	**3.77**	0.49	**9.05**	91.56	**8.48**	0.39
EDDA$_{\text{SA}}$	55.84	84.19	1.84	0.52	35.03	**83.19**	4.68	**1.22**

(b) PASCAL VOC 2012

Model	ResNet-50				VGG-16			
Metric	Drop		Increase		Drop		Increase	
f	GC	SA	GC	SA	GC	SA	GC	SA
No Aug	46.70	96.13	0.39	0.01	43.90	91.28	0.83	0.13
CutMix	57.95	96.20	2.11	**0.07**	72.55	96.19	1.09	0.00
MixUp	53.80	95.18	**2.51**	0.02	53.83	95.13	**1.80**	0.02
EDDA$_{\text{GC}}$	**29.53**	94.94	1.45	0.00	**21.88**	91.99	1.68	0.07
EDDA$_{\text{SA}}$	43.50	**92.74**	0.72	0.04	44.12	**77.02**	0.83	**0.28**

(c) Oxford-IIIT Pet Dataset

Table 2: ResNet-50 and VGG-16 model performance in terms of Drop%$_{\text{mag}}$ and Increase%$_{\text{mag}}$ on CIFAR-100, PASACL VOC 2012, and Oxford-IIIT Pet Dataset. We report the performance of both employing Grad-CAM (GC) and Saliency Map (SA) as the saliency method f used on the threshold function $T(\cdot)$ when calculating Drop% and Increase%. For our method, we report the performance of the models that trained using GC and SA as the explainers. We compare the performance of our method with CutMix, MixUp, as well as no augmentation (No Aug).

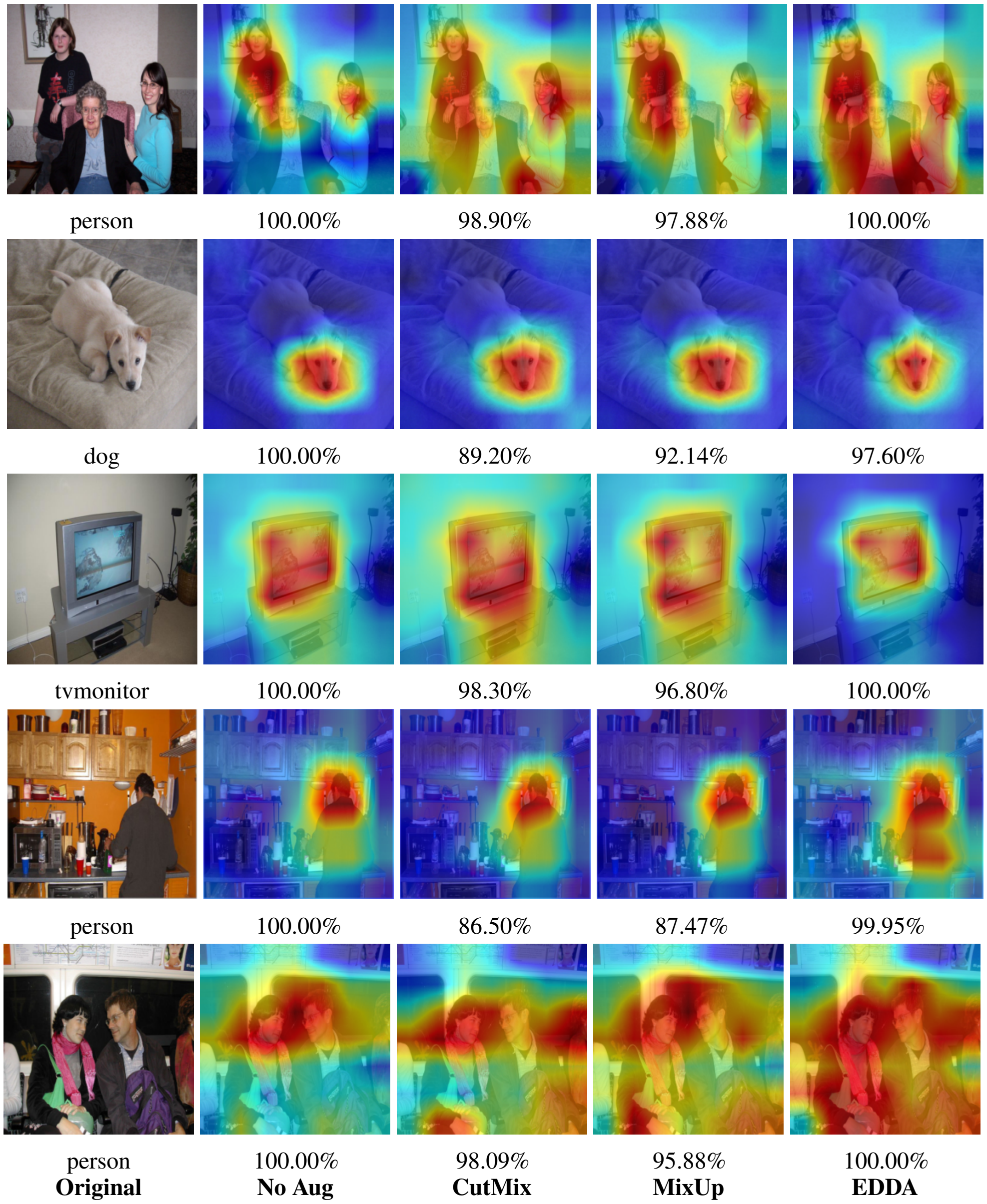

Figure 3: Some more examples of heatmap visualizations. We also included the confidence score for the prediction under each method. Our augmentation method EDDA always got comparable confidence scores compared to no augmentation, higher than both CutMix and MixUp.