Received December 18, 2020, accepted January 6, 2021, date of publication January 11, 2021, date of current version January 27, 2021.

*Digital Object Identifier 10.1109/ACCESS.2021.3050758*

# Local Augment: Utilizing Local Bias Property of Convolutional Neural Networks for Data Augmentation

**YOUMIN KIM, A. F. M. SHAHAB UDDIN, AND SUNG-HO BAE, (Member, IEEE)**
Department of Computer Science and Engineering, Kyung Hee University, Yongin 17104, Republic of Korea

Corresponding author: Sung-Ho Bae (shbae@khu.ac.kr)

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program (IITP-2019-2015-0-00742) supervised by the IITP (Institute of Information & Communications Technology Planning & Evaluation).

**ABSTRACT** Data augmentation is an effective way to increase the diversity of existing training datasets that result in improved generalization ability of convolutional neural networks (CNNs). The augmentation effect is usually global for the existing methods i.e., a single augmentation effect is applied to the whole image, thus limiting the diversity of local characteristics in augmented images. Moreover, the global augmentation effect does not support the most fundamental behavior of CNNs i.e., they focus more on local features (local texture, tiny noise etc.) than global shapes. We refer to this behavior as local bias property. In this paper, we propose a new data augmentation method, called Local Augment (LA), which highly alters the local bias property so that it can generate significantly diverse augmented images and offers the network with a better augmentation effect. First, we select few local patches in an image, then apply different types of augmentation strategies to each local patch. This augmentation process collapses the global structure of the object but creates locally diversified samples, which helps the network to learn the local bias property in a more generalized way. As a result, it increases the generalizability and the prediction accuracy of the network. To verify the effectiveness of the proposed method, we perform comprehensive experiments on image classification with benchmark datasets, where the proposed method outperforms the sate-of-the-art data augmentation techniques on ImageNet and STL10 and shows competitive performance on CIFAR100.

**INDEX TERMS** Image classification, overfitting, data augmentation, local bias property, multiple augmentation effects.

## I. INTRODUCTION

Convolutional neural networks (CNN) have shown promising results in almost every field [6], [16], [18], [19], [22], [23] due to their complex feature representation ability. However, they have a trend to be overfitted when the training data is insufficient i.e., the model parameters excessively fit to the training data. As a result, they poorly perform on unseen data (test data) [5]. To solve this problem, various methods e.g., dropout [1], [25], network ensemble [9], [17] and data augmentation [21], [26] have been proposed. Dropout and network ensemble solve the overfitting problem by manipulating the network architecture and/or its weights, while the data augmentation solves the problem by directly manipulating data distribution. This paper focuses on data augmentation methods.

Data augmentation [21], [26] enriches the diversity of data by creating new samples with the help of some transformations applied to the original training data. As a result, these diversified data can prevent the network from being overfitted to the original training data. In general, traditional augmentations apply linear transformations (shifting, rotation, flipping, shearing and etc.) to the existing training data to create new samples that help to change the distribution of the data. Recent works [8], [30], [32], [33] have made the distribution of the original training data more diverse by applying nonlinear transformations, such as zeroing out local regions in an image, injecting random noise, combining two different images and so on. Thanks to the high flexibility of nonlinear operations, data augmentation methods

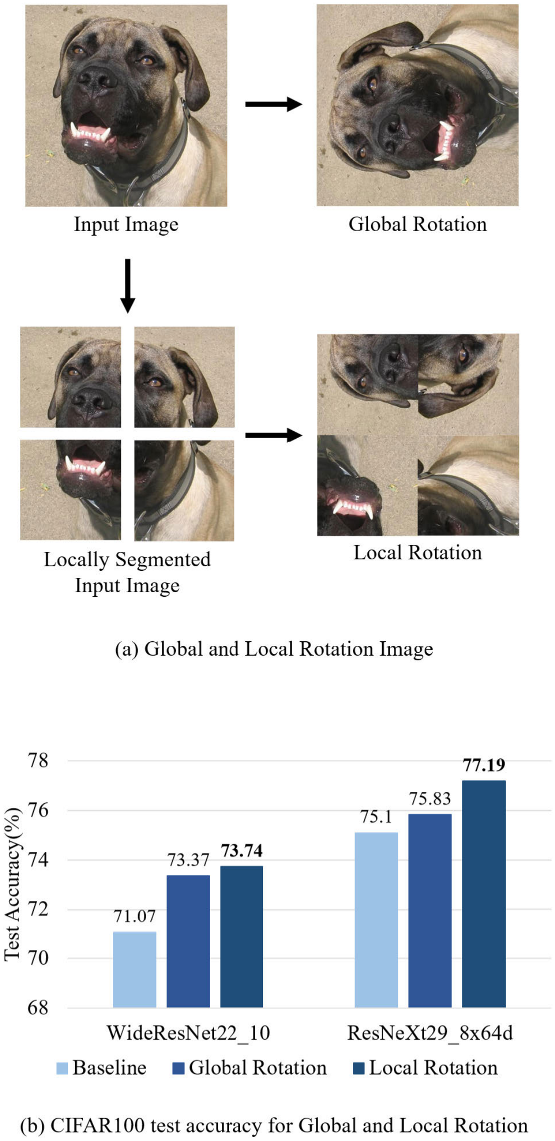
The associate editor coordinating the review of this manuscript and approving it for publication was Shadi Alawneh.

with non-linear operations tend to have higher generalization performance compared to the augmentation methods with linear operations [8], [30], [32], [33].

The augmentation methods with nonlinear operations [8], [30], [32], [33] are divided into two groups: intra-image and inter-image augmentation methods. The intra-image augmentation methods create images by applying transformation on a single image source, where the labels of the created images remain the same with the source image. On the other hand, inter-image augmentation methods create images by combining multiple image sources where the labels of the created images are made by interpolation among the labels of source images. Due to the unlimited number of possible combinations and label smoothing effect, the inter-image augmentation method have shown higher generalization performance compared to the intra-image augmentation when being used in CNNs [30], [32].

However, the inter-image augmentation methods can suffer from the problem of mixing unimportant image regions (e.g., background) when generating new samples, resulting in performance degradation of CNNs. For example, in Mixup [32], a new image is created by combining the two images with different transparencies, thus leading to generating unnatural image structures as mentioned in CutMix [30]. Instead of mixing two different images, CutMix [30] cuts a patch from an image and then pastes it to another image to generate a new sample. However, CutMix [30] also may suffer from mixing a meaningless patch or a patch which is irrelevant to the interpolated object label [27]. In addition, the conventional augmentation methods are limited to applying a single augmentation effect on an image which globally alters the image characteristics. We refer to this fact as global augmentation effect.

In order to overcome the aforementioned problems, we propose a new augmentation method where we select local patches in an image, and then apply different augmentation effects to each local patch to generate a new sample. each local image region takes multiple augmentation effects so that a much variety of images can be created for training CNNs. Theoretically, our proposed augmentation method relies on the local bias property which is a fundamental behavior of CNN [2]–[4], [11], [24], [28]. Recent studies [2]–[4], [11], [24], [28] have found that, in image classification task, CNN is greatly influenced even by a small noise since it classifies an input image based on the local shape and texture rather than the global structure of the object. This characteristic is considered as a disadvantage of CNNs and difficult to overcome. However, our method have considered this characteristic as a beneficial tool for data augmentation such that it creates new samples containing locally diversified augmentation effects.



(a) Global and Local Rotation Image

(b) CIFAR100 test accuracy for Global and Local Rotation

**FIGURE 1. Comparisons with a global and local augmentation method. (a) Images with global and local data augmentation method (rotation). (b) CIFAR100 test dataset accuracy results from baseline, global rotation and local rotation for WideResNet22_10 [31] and ResNeXt29_8 × 64d [29].**

We conduct a simple experiment to see the effectiveness of the local bias property in data augmentation. Figure 1 compares the performance in terms of test accuracy for the data augmentation with global and local image rotations on CIFAR100. Figure 1(a) visualizes the augmented images and Figure 1(b) shows the performance comparison of global and local augmentation where the two baseline networks are WideResNet [31] and ResNext [29]. Note that all experimental specifications are identical to Section III.A.

As shown in Figure 1(b), applying both the global and local image rotations yields higher test accuracy compared to the baseline models that do not use data augmentation. Further the local image rotation outperforms the global image rotation in both baseline networks.

These experimental results imply that locally variant data augmentation methods can offer higher generalization performance to a network due to the local bias property of CNNs and higher flexibility of data augmentation methods. Based on this observation, we propose a new data augmentation method, Local Augment (LA), which selects local

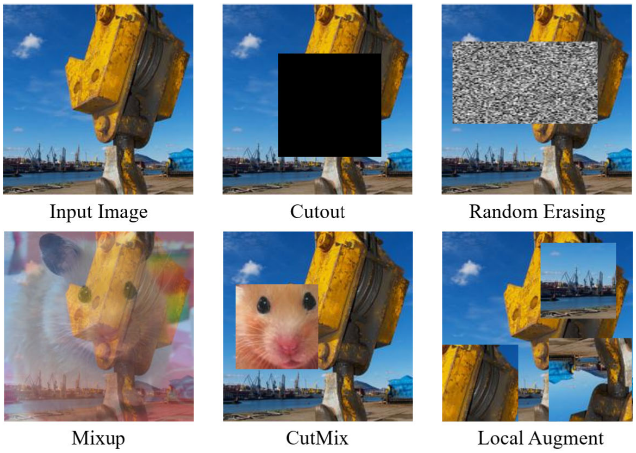


**FIGURE 2. Images with various data augmentation methods: (from top-left) Input image, Cutout [8], Random Erasing [33], Mixup [32], CutMix [30] and our LA.**

patches in an image, and then applies different augmentation effects (flipping, rotation and channel-shuffling) to each local patch. Comprehensive experimental results demonstrate the superiority of our method compared to the conventional global data augmentation methods on benchmark datasets in image classification task.

The main contribution of this paper are as follows:

- To demonstrate the local bias property of CNNs, we train a network using traditional data augmentation with destructed global shape of an object. And we reveal that utilizing this property can improve the model performance.
- Based on the above mentioned evidence (i.e., the local bias property) we propose a new data augmentation method, called Local Augment (LA), which destructs the global shape and applies various augmentation effects to the local image patches.
- Our method outperforms the state-of-the-art data augmentation methods on ImageNet and STL10 classification tasks and shows competitive performance on the CIFAR100 classification task.

## II. RELATED WORK

### A. DATA AUGMENTATION

Data augmentation is one of the effective regularization techniques that aims to prevent overfitting of a network and increases the generalization performance [8], [30], [32], [33]. The data augmentation techniques create more affluent training data, transformed from the original such that the trained network gains higher generalization performance to unseen test data.

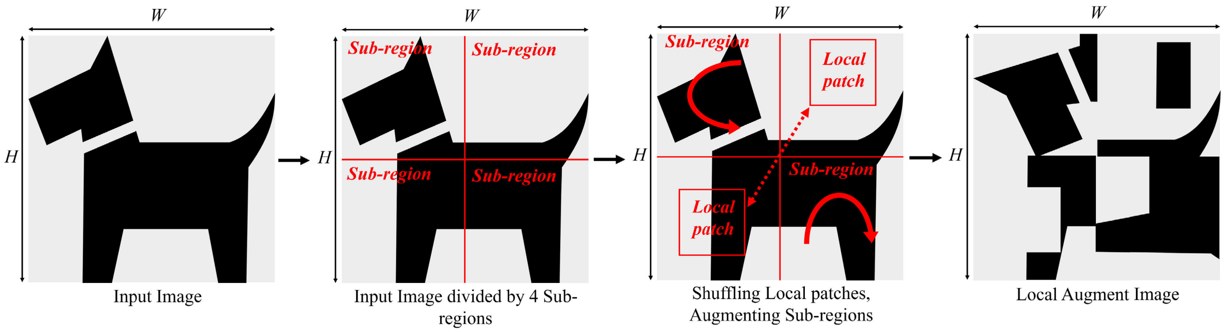
There have been many transformation techniques to create a new image in previous data augmentation methods. Figure 2 illustrates some representative data augmentation methods, i.e., Cutout [8], Random Erasing [33], Mixup [32], CutMix [30], and our LA. In [8], [33], to create a new training image, a certain part of an original input image is erased with zero values (Cutout) or replaced with random values (Random Erasing) which can be considered dropout effects [25] on the input data. In [30], [32], two different original images are used to create a new image. In Mixup [32], the two different original images are interpolated with random ratio at the same pixel position to create a blended image. Also, to assign a new label data (one-hot-encoded vector) for the new created image, the label data for the two images are blended by a element-wise vector interpolation. In CutMix [30], a certain part of an original input image is randomly replaced with a certain part of another original image, where the label vector for the new created image is made with the element-wise vector interpolation with the two label data corresponding to the each image. The ratio for the vector interpolation is determined by areas occupied by the two original images in the new image.

Although these aforementioned data augmentation methods have shown promising generalization performance, all of them are global augmentation methods, limited to have only one augmentation effect in a new image. Unlike these methods, our method can take advantages of numerous augmentation effects by applying different augmentation strategies on each local part in the new image.

### B. LOCAL BIAS PROPERTY

Recently, many studies have revealed that CNN is biased to local features (textures, tiny noises and etc) and it has been considered as a disadvantage for improving the network robustness. In [4], the image-independent universal patch is developed to apply adversarial attack on neural networks and easily fools the network to wrongly classify the image in image classification tasks [12], [20]. In [2] and [11], researchers have found the local bias property by confirming the wrong predictions from inconsistent input images where the local texture of the images and global shape of the objects in the images are not consistent. Especially in [11], the AdaIN [14] style transfer dataset where texture of object is changed to different paintings is created to train the network to be less biased to the local texture and more biased to the global shape. Through this, the robustness of the network is improved by supressing the local bias property.

In [3], it is revealed that, for training networks, using only several local image patches of input images can bring out similar performance compared to the network trained with the whole image regions in ImageNet classification task [23]. Through this, they reveal that the behavior of CNN is still similar to bag-of-feature models [10] that use only local features for classification tasks. In [28], an additional classifier that takes the local features of an image as input is trained to make wrong prediction and the main classifier that takes the global features of the image as input is trained to make right prediction. This method allows the network to take full consideration for global representation while suppressing the dependency on local representation to enhance the robustness for adversarial attack, thus achieving good performance in various domain datasets. In [24], the global structure bias, a disadvantage in adversarial training is overcome by training



Input Image

Input Image divided by 4 Sub-regions

Shuffling Local patches, Augmenting Sub-regions

Local Augment Image

**FIGURE 3. Local Augment total process. First, we divide an input image by 4 sub-regions. Second, we select local patches to be shuffled and sub-regions to be augmented. Finally, we shuffle local patches and augment sub-regions.**

**TABLE 1. CIFAR100 test dataset Top1 accuracy results with baseline and global/local data augmentation for WideResNet22_10 [31] and ResNeXt29_8 × 64d [29]. The best case between the global and local methods is shown through the check mark for each augmentation method.**

| Network | | Top1 Acc(%) | Best Case |
|---|---|---|---|
| WideResNet (Baseline) | | 71.07 | - |
| + Rotation | + Global | 73.37 | |
| | + Local | 73.74 | ✓ |
| + Flipping | + Global | 76.94 | ✓ |
| | + Local | 75.13 | |
| + Rotation & Flipping | + Globlal | 76.00 | ✓ |
| | + Local | 75.84 | |
| ResNeXt (Baseline) | | 75.10 | - |
| + Rotation | + Global | 75.83 | |
| | + Local | 77.19 | ✓ |
| + Flipping | + Global | 79.24 | ✓ |
| | + Local | 77.20 | |
| + Rotation & Flipping | + Globlal | 76.60 | |
| | + Local | 76.73 | ✓ |

a network to make robust local features through Random Block Shuffle in adversarial training methods [12], [20].

## III. METHOD

### A. PRELIMINARY EXPERIMENTS ON EFFECTIVENESS OF LOCAL DATA AUGMENTATION

Before designing the proposed method, we conduct a simple experiment to see if CNN's local bias property helps to improve data augmentation performance. We consider two cases of augmentation: (i) data augmentation on the whole image (global augmentation effect) and (ii) data augmentation on local patches. Then we compare their effect in terms of test accuracy. We train WideResNet [31] and ResNeXt [29] on CIFAR100 dataset and use the rotation and flipping for data augmentation in the image classification task.

Figure 1 shows the case of global and local data augmentation. For local augmentation, it can be seen that even if we use a single augmentation effect, it can be applied on each local patches differently e.g., rotation with 90°, 180° and 270°, flipping with horizontal and vertical directions that allows one image to have several augmentation effects. Although this causes the global structure to be collapsed, it offers the network with better performance. In other words, this experiment shows whether a network has a greater dependence on the global structure of an object or on various local information of an image during data augmentation.

Table 1 presents the experimental results. It can be seen that when flipping is applied to an image, both networks show lower performance when the local augmentation method is used compared to the global augmentation method. On the other hand, local augmentation method shows higher performance when rotation is applied. Furthermore, when rotation and flipping are applied together, the two networks show comparable performance regardless of local and global augmentation methods. These experimental results imply that that applying appropriate augmentation strategy on each local image patch may improve the generalization ability of a network, despite the global shape collapsing.

### B. PROPOSED LOCAL AUGMENT

Based on the above discovery, we propose Local Augment (LA) to diversify the augmentation effect of an input image using CNN's local bias property. Before explaining a total process of the proposed method, we define sub-region and local patch. The sub-region accounts for a local part of an input image and the local patch is in the sub-region. Figure 3 shows the total process of the proposed method. First, we divide the original image into $N$ sub-regions. Empirically, we set $N$ as 4 which is identically applied to all the experiments. Second, we perform different data augmentation techniques in a unit of sub-region. Especially, we also propose to use a local patch which is defined in a sub-region where we perform shuffling between two local patches. The number of the local patches are randomly selected in the uniform distribution ranging from 0 to 4. After shuffling the local patches, LA performs data augmentation with a 50% probability for the remaining sub-regions which do not shuffle the local patches. That is, each rotation/flipping mode (among 90°, 180° and 270°rotations and horizontal and vertical direction flipping) has 10% probability for being selected. After that, a channel shuffling is applied to all the augmented sub-regions and shuffled with a 25% probability in the RGB domain. In Section V, we provide empirical bases for the aforementioned carefully designed augmentation strategy.

Figure 4 shows how the local patch is selected in the sub-region, a local part of an input image. In this paper, we create the sub-regions by dividing width and height of the original image $(W, H)$ into two halves $(W/2, H/2)$ as $N = 4$. The local patch is selected from a random location $(b_w, b_h)$ in the sub-region. The location $(b_w, b_h)$ and size of the local patch $(l_w, l_h)$ are set by a border value $B$ which is set to be 10% of the width and height size for the input image which. The width and height in the local patch is expressed as:

$$l_w = \frac{W}{2} - B$$
$$l_h = \frac{H}{2} - B \qquad (1)$$

The location $(b_w, b_h)$ is sampled with a uniform distribution parameterized by $B$ as

$$b_w, b_h \sim U(0, B) \qquad (2)$$

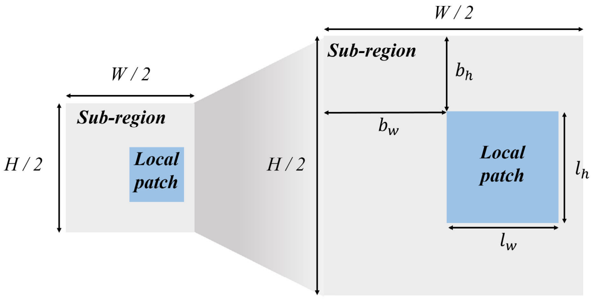


**FIGURE 4.** Definition of sub-region and local patch. The size and location of a local patch is randomly selected in a sub-region.

## IV. EXPERIMENTS

We conduct image classification experiments on three datasets i.e., ImageNet [16], STL10 [7], CIFAR100 [15] to compare the proposed LA with other representative augmentation methods [8], [30], [32], [33]. Except ImageNet, all the experiments are conducted three times and their average values taken for comparison. Since the image dimension and number of classes vary from dataset to dataset, and the augmentation methods are applied with a certain probability (we call this probability as *method probability* throughout the paper) for each mini-batch during the training process, a few hyper-parameters are required. All of these hyper-parameters are carefully selected either by following the original papers, or based on our experimental observations. The method probability for Cutout [8] and RandomErasing [33] is 1.0 and 0.5 respectively. Cutout mask size is the same as used in the original paper [8] for CIFAR100 and we manually set the size 24 × 24 for STL10, 112 × 112 for ImageNet datasets. In RandomErasing [33], we set erased rectangle area scale from 0.02 to 0.4 and ratio $r_1, r_2$ as 0.3, 3.3, respectively for CIFAR100 and STL10 datasets. The $\alpha$ for Mixup [32] is set to 1.0. The method probability of CutMix [30] is set to 0.5, 0.5 and 1.0 for CIFAR100, STL10 and ImageNet datasets, respectively. The method probability of the proposed method is set to 0.5 for all datasets and hyper-parameter $B$ is set to 3, 8 and 22 for CIFAR100, STL10 and ImageNet, respectively.

### *A. ImageNet*

ImageNet dataset [23] consists of 1.28M training images and 50K test images from 1K different classes. In training phase, we apply standard augmentations such as resizing to 256 × 256 pixels, random cropping to 224 × 224 pixels and random horizontal flipping. The networks are trained for 100 epochs with a batch size of 256. We use SGD optimizer with learning rate of 0.1, momentum of 0.9 and weight decay of 0.0001. The learning rate is decayed by a factor of 0.1 after each 30, 60 and 90 epochs. In test phase, We use resizing to 256 × 256 pixels and center cropping to 224 × 224 pixels. We use ResNet50 [13] as a baseline network for this experiment. Table 2 presents the experimental results. We can see that the LA achieves top-1 accuracy of 76.87% that outperforms the state-of-the-art (SOTA) methods under comparison. Moreover, the test accuracy of the proposed method is 0.05% and 0.37% higher than CutMix [30] and Mixup [32], respectively.

**TABLE 2.** ImageNet test dataset Top1 and Top5 accuracy results with local augment (LA) and other augmentation methods for ResNet50 [13].

| Network | Top1 Acc(%) | Top5 Acc(%) |
|---|---|---|
| ResNet50 (Baseline) | 76.09 | 92.93 |
| + Cutout [8] | 76.24 | 93.06 |
| + Mixup [32] | 75.50 | 92.82 |
| + CutMix [30] | 76.83 | **93.42** |
| + LA (Ours) | **76.87** | 93.35 |

Note that, in addition to global augmentation, CutMix and Mixup exploit label smoothing technique that improves the performance significantly. Compared to CutMix and Mixup, the proposed LA is an intra augmentation method which cannot adopt label smoothing techniques. Therefore, our experimental results imply that applying augmentation techniques to local regions may generate images with higher diversity and quality compared to global augmentation techniques.

Figure 5 shows Class Activation Map (CAM) [34] which focuses on the label-related portion of the test image for networks trained by each augmentation method. The object name in the test image of Figure 5 is *Haematopus ostralegus*, which features black head and long red beaks and legs. The CAM for the test image shows that CutMix [30] captures a more global area than other augmentation methods. However, it can be seen that the proposed LA has higher performance even though it captures more local parts than CutMix [30]. Compared to Cutout [8] and Mixup [32], which captures relatively more local parts except for CutMix [30], Local Augment captures more meaningful local parts. Cutout [8] captures the head and beak well, but only for one of the two birds. Mixup [32] catches both the two birds, but fails to capture the features that express the birds well (black head, red beak) and captures meaningless or trivial features (white boats). But our method captures the most representative feature for both of the two birds.

This experimental results show that our method which has multiple augmentation effects using local bias property

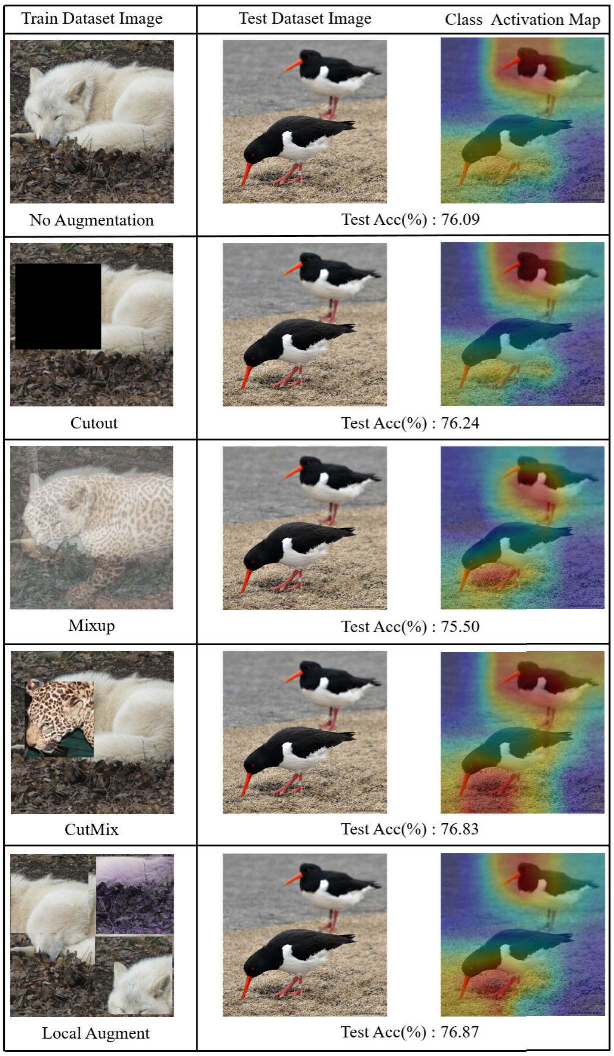


**FIGURE 5.** Results of ImageNet classification on no augmentation, various augmentation methods [8], [30], [32], [33] and Local Augment. Class Activation Map [34] shows where each ResNet50 [13] pre-trained by each augmentation methods focuses on. Catching high quality of local information in Local Augment is just enough to improve classification performance and outperform all augmentation methods.

learns important local features of an object. And in summary, it suggests that not only capturing global feature of an object improves the generalization performance in image classification task but also capturing the most representative local features increase the generalization performance.

### B. STL10

STL10 [7] dataset consists of labeled data and unlabeled data. The labeled data consists of 500 training images and 800 test images for 10 different classes, and the unlabeled data consists of 100k images. All the images are of 96 × 96 pixels with RGB format. This dataset is widely used for unsupervised learning and it has a small number of labeled data. However, we use this dataset to show that our method also works well even when the number of training data is significantly small. In this experiment, we train each of networks for 1000 epochs with a batch size of 64. Before using the training images, we apply random crop from the images with 4 pixels padding size and horizontal flipping with 0.5 probability. We use SGD as optimizer with learning rate of 0.1, momentum of 0.9 and weight decay 0.0005. Learning rate is decayed by a factor of 0.2 after 300, 400, 600 and 800 epochs. We use WideResNet [31] with a depth of 22 and a width factor of 10 and ResNeXt [29] with a depth of 29, a cardinality factor of 1 and a width factor of 64 as baseline networks for this experiment.

**TABLE 3.** STL10 test dataset Top1 accuracy results with Local Augment(LA) and other augmentation methods for WideResNet22_10 [31] and ResNeXt29_1 × 64d [29]. 'LS' denotes the label smoothing in CutMix [30].

| Network | Top1 Acc(%) | Increased Acc(%) |
|---|---|---|
| WideResNet (Baseline) | 82.53 | - |
| + Cutout [8] | 80.86 | (-1.67) |
| + RandomErasing [33] | 67.61 | (-14.92) |
| + Mixup [32] | 86.64 | (+4.11) |
| + CutMix [30] (w/o LS) | 83.99 | (+1.46) |
| + CutMix [30] | 87.93 | (+5.40) |
| + LA (Ours) | **88.65** | **(+6.12)** |
| + LA (Ours) + CutMix [30] | **89.80** | **(+7.27)** |
| ResNeXt (Baseline) | 84.00 | - |
| + Cutout [8] | 83.41 | (-0.59) |
| + RandomErasing [33] | 70.09 | (-13.91) |
| + Mixup [32] | 85.68 | (+1.68) |
| + CutMix [30] (w/o LS) | 81.58 | (-2.42) |
| + CutMix [30] | 86.05 | (+2.05) |
| + LA (Ours) | **86.33** | **(+2.33)** |
| + LA (Ours) + CutMix [30] | **86.92** | **(+2.92)** |

Table 3 shows the experimental results of the data augmentation methods in comparison for STL10 image classification. Local Augment achieves top-1 accuracy of 88.65% and 86.33% for WideResNet [31] and ResNeXt [29] which outperforms other state-of-the-art (SOTA) methods although they use multiple images with label smoothing. In addition, when the proposed method is combined with CutMix [30], the performance get further increased. In Cutout [8], Random Erasing [33] and the proposed method, a new sample is created by using a single image. However Cutout [8] and Random Erasing [33] decrease the baseline performance, while the proposed method significantly improves the network performance. It can be seen that the locally diversified augmentation effects in the proposed method highly contribute to the performance improvement.

### C. CIFAR100

CIFAR100 dataset consists of 60K images for 100 classes, where 50K are used as training set and 10K are used as test set. Each set has RGB images of 32 × 32 pixels. Here, we train all of the networks for 200 epochs with a batch size of 64. Besides the data augmentations that are in comparison, we apply random crop on original images with 4 pixels padding size and horizontal flip with 0.5 probability. SGD is used as an optimizer with learning rate of 0.1, momentum of 0.9 and weight decay of 0.0005. Learning rate is decayed by a factor of 0.1 after each 100 and 150 epochs. We use WideResNet [31] with a depth of 22 and a width factor

**TABLE 4. CIFAR100 test dataset Top1 and Top5 accuracy results with Local Augment(LA) and other augmentation methods for WideResNet22_10 [31] and ResNeXt29_8 × 64d [29]. 'LS' denotes the label smoothing in CutMix [30].**

| Netowrk | Top1 Acc(%) | Top5 Acc(%) |
|---|---|---|
| WideResNet (Baseline) | 80.46 | 95.01 |
| + Cutout [8] | 80.68 | 95.32 |
| + RandomErasing [33] | 80.91 | 95.34 |
| + Mixup [32] | 81.40 | 95.39 |
| + CutMix [30] (w/o LS) | 81.35 | 95.69 |
| + CutMix [30] | 82.38 | 95.91 |
| + LA (Ours) | **81.81** | **95.92** |
| + LA (Ours) + CutMix [30] | **82.68** | **96.34** |
| ResNeXt (Baseline) | 80.81 | 95.38 |
| + Cutout [8] | 80.98 | 95.34 |
| + RandomErasing [33] | 80.60 | 95.48 |
| + Mixup [32] | 81.60 | 95.06 |
| + CutMix [30] (w/o LS) | 80.81 | 95.74 |
| + CutMix [30] | 82.04 | 95.80 |
| + LA (Ours) | **81.61** | **96.15** |
| + LA (Ours) + CutMix [30] | **82.30** | **96.15** |

of 10 and ResNeXt [29] with a depth of 29, a cardinality factor of 8 and a width factor of 64 as baseline networks for this experiment. Table 4 shows the experimental results of the data augmentation methods in comparison for CIFAR100 image classification. Local Augment shows the highest top-1 and top-5 accuracy among the augmentation methods that only uses a single image for augmentation. And the accuracy is higher than Mixup [32], which uses multiple images. Also, when the proposed method is combined with CutMix [30], it outperforms all other methods.

Since the proposed method only uses a single image, it improves the performance of the network even though there is no label smoothing effect. This is why the proposed method results in better overall performance than CutMix [30] in Table 2 and Table 3, but not all of them (Table 4). However, when combining CutMix and LA, it shows better performance than CutMix because the proposed method obtains the label smoothing effect as well. Therefore, we performed the experiments with CutMix+LA on CIFAR-100 to show that the performance is further improved. In addition, by removing label interpolation from CutMix (e.g., the label for a created image is identically set to the label of the target image), the CutMix without label smoothing shows lower performance than the proposed method. Through this, it can be concluded that the label smoothing effect contributed to the performance improvement of CutMix significantly.

## V. ABLATION STUDIES

In this section we investigate how Local Augment helps a network to learn significant feature representation during training and also analyze several components and degree of the proposed method.

### 1) HOW DOES LOCAL AUGMENT TRAIN A NEURAL NETWORK

To find out how the augmented images with Local Augment affects a network, we visualize the Class Activation Map (CAM) [34] of ResNet50 [13] which is trained by using Local Augment on ImageNet. The third and fourth row of Figure 6 shows experimental results for baseline ResNet50 [13], pre-trained without augmentation and with our method. In baseline, when global structure is collapsed by our method, it fails to focus on the most representative local information of the object in the image. But when trained with the proposed data augmentation method, it focuses well on the significant local part of the object. For *Tench* and *Beaker* image, the baseline network fails to capture the local part of an object when collapsed by the shuffling operation and focuses on meaningless information in the center. But the network trained with Local Augment correctly focuses on most important local parts. In *Hamster* image, baseline focus on global structure even when the object is collapsed, but ours focuses on a certain local part. In *Chocolate Sauce* image, baseline focuses on the trivial local information (neck part that does not have much sauce), whereas ours focuses on the core local information (forehead part where the sauce is concentrated).

Local Augment image of Figure 6 shows that shuffling operation makes a network distracted by destruction of object's global structure in an original image. However, if we use these destructed images to train the network without changing label of the image, it rather increases the generalization ability of the network by allowing the network to learn more robust local features by utilizing locally diversified augmentation effects.

### 2) COMPONENTS AND DEGREES OF LOCAL AUGMENT

The shuffling operation of our method consists of spatial-shuffling and channel-shuffling. We perform experiments to find out how each component contributes to the performance improvement. Here we perform experiments on CIFAR100 using ResNet56 [13] and all the configurations are kept the same as mentioned in Section IV. The experimental results are presented in Table 5. It can be seen that spatial-shuffling for local patches rather than sub-regions significantly improves the model performance. This can be seen as an evidence that varying the local patches through the border value $B$, helps diversifying the augmentation effect. In addition, when channel-shuffling is added, the local patch shuffling drops its performance slightly, while the sub-region shuffling improves its performance by a large margin. Therefore, for generalization of our method, we include channel-shuffling in the proposed method.

To find out how the degree of our method in each mini-batch and channel-shuffling contribute to the performance improvement, we perform experiments with varying the method probability and channel-shuffling probability and other configurations are kept the same as mentioned in Table 5. As Table 6 shows, the greater the degree of Local Augment in a mini-batch, the more generalization ability the network can achieve. But the generalization performance is decreased when the degree of channel-shuffling increases. This supports validity of data augmentation effect of our

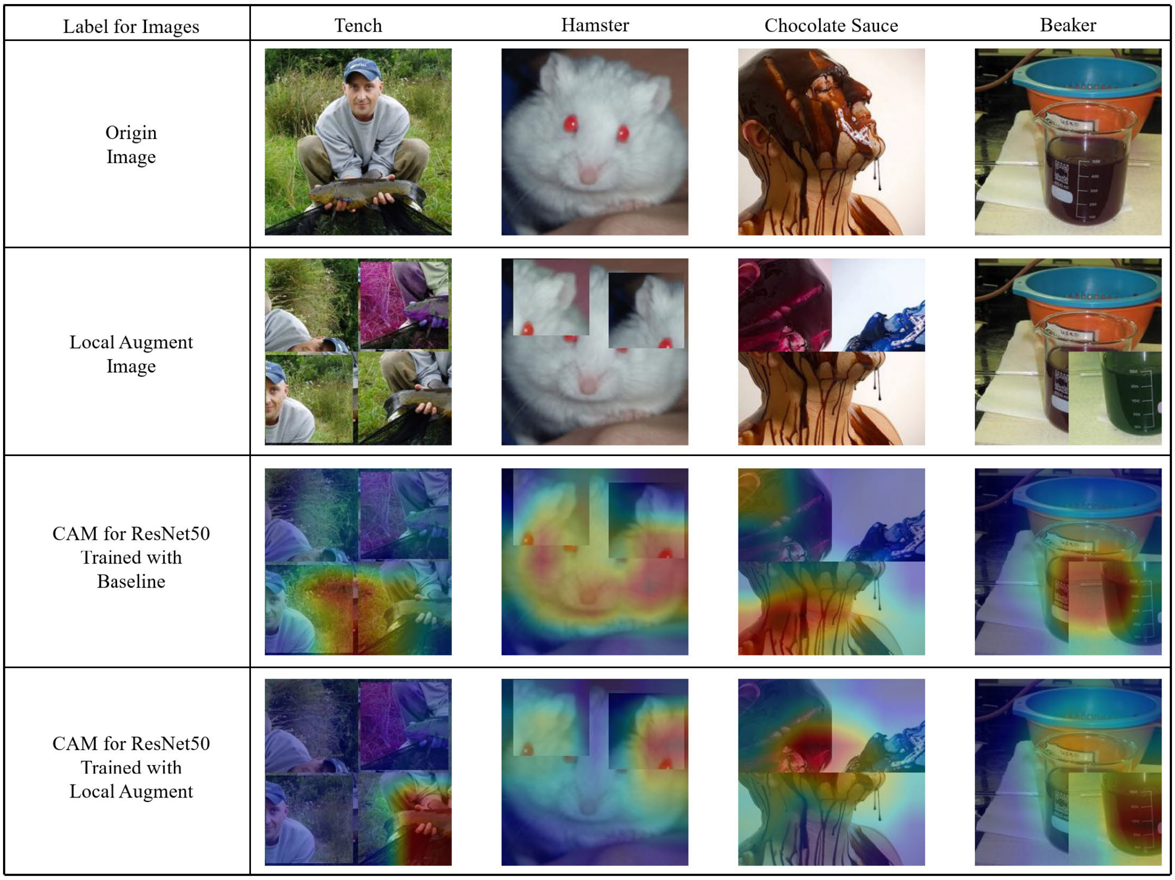

Label for Images
Tench
Hamster
Chocolate Sauce
Beaker
Origin Image
Local Augment Image
CAM for ResNet50 Trained with Baseline
CAM for ResNet50 Trained with Local Augment

**FIGURE 6.** Class Activation Map (CAM) [34] for Local Augment images. We compared a baseline trained network and Local Augment trained network. Through these maps, we can find out where each network trained with baseline and our method focuses on in the input images.

**TABLE 5.** CIFAR100 test dataset Top1 accuracy results with baseline, CutMix [30] and our various component versions for ResNet56 [13].

| Network | Top1 Acc(%) |
|---|---|
| ResNet56 (Baseline) | 72.02 |
| + CutMix [30] | 74.60 |
| + Sub-Region Mixing | 73.71 |
| + Sub-Region Mixing + Channel Shuffling | 74.32 |
| + Local Patch Mixing | 74.49 |
| + Local Patch Mixing + Channel Shuffling | 74.28 |

**TABLE 6.** CIFAR100 test dataset Top1 accuracy results with various degrees of Local Augment (LA) and channel shuffling for ResNet56 [13].

| Probability | | Per Samples | | |
|---|---|---|---|---|
| | | 0.50 | 0.75 | 1 |
| Channel Shuffling | 0 | 74.49 | 74.42 | 74.60 |
| | 0.25 | 74.28 | 74.32 | **74.97** |
| | 0.50 | 73.88 | 74.43 | **74.73** |
| | 0.75 | 73.79 | 74.52 | 74.49 |

method and shows that contribution of channel-shuffling should be small. When we set the method probability to 1.0 and channel shuffling probability to 0.25 or 0.5, our method outperforms CutMix [30], state-of-the-art method which uses multiple images during data augmentation.

## VI. CONCLUSION

In this paper, we present a new method called Local Augment by utilizing the local bias property of a convolutional neural network i.e., the tendency to focus on local part (textures, tiny noises and etc) of an input image. Our method overcomes the shortcomings of an unrecognizable object in an input image due to global structure collapsing of the object through locally diversified augmentation effect, and improves the generalization performance of a network. Local Augment uses only a single image to create a new sample with multiple augmentation effects. Extensive experiments on image classification for several benchmark datasets and popular network architectures validates the excellence of the proposed data augmentation method. Local Augment provides a new insight into designing a novel data augmentation method by discovering that the local bias property can be used as a beneficial tool for improving the generalization performance of a network.

## ACKNOWLEDGMENT

The authors are very grateful for Eunseop Shin for technical support.

## REFERENCES

[1] J. Ba and B. Frey, "Adaptive dropout for training deep neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3084–3092.

[2] N. Baker, H. Lu, G. Erlikhman, and P. J. Kellman, "Deep convolutional networks do not classify based on global object shape," *PLOS Comput. Biol.*, vol. 14, no. 12, Dec. 2018, Art. no. e1006613.

[3] W. Brendel and M. Bethge, "Approximating CNNs with bag-of-local-features models works surprisingly well on ImageNet," 2019, *arXiv:1904.00760*. [Online]. Available: http://arxiv.org/abs/1904.00760

[4] T. B. Brown, D. Mané, A. Roy, M. Abadi, and J. Gilmer, "Adversarial patch," 2017, *arXiv:1712.09665*. [Online]. Available: http://arxiv.org/abs/1712.09665

[5] R. Caruana, S. Lawrence, and C. L. Giles, "Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 402–408.

[6] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[7] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 215–223.

[8] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," 2017, *arXiv:1708.04552*. [Online]. Available: http://arxiv.org/abs/1708.04552

[9] T. G. Dietterich, "Ensemble methods in machine learning," in *Proc. Int. Workshop Multiple Classifier Syst.* Springer, 2000, pp. 1–15. [Online]. Available: https://link.springer.com/chapter/10.1007/3-540-45014-9_1

[10] L. Fei-Fei, R. Fergus, and A. Torralba, "Recognizing and learning object categories: Part 1: Bag of words models," in *Proc. ICCV Short Course*, 2005.

[11] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness," 2018, *arXiv:1811.12231*. [Online]. Available: http://arxiv.org/abs/1811.12231

[12] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2014, *arXiv:1412.6572*. [Online]. Available: http://arxiv.org/abs/1412.6572

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[14] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1501–1510.

[15] A. Krizhevsky, V. Nair, and G. Hinton, "Learning multiple layers of features from tiny images," Cifar-10 (Canadian Institute for Advanced Research), Tech. Rep., ch. 3.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[17] A. Krogh and J. Vedelsby, "Neural network ensembles, cross validation, and active learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 1995, pp. 231–238.

[18] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 21–37. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-46448-0_2

[19] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[20] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2017, *arXiv:1706.06083*. [Online]. Available: http://arxiv.org/abs/1706.06083

[21] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," 2017, *arXiv:1712.04621*. [Online]. Available: http://arxiv.org/abs/1712.04621

[22] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767

[23] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[24] C. Song, K. He, J. Lin, L. Wang, and J. E. Hopcroft, "Robust local features for improving the generalization of adversarial training," 2019, *arXiv:1909.10147*. [Online]. Available: http://arxiv.org/abs/1909.10147

[25] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[26] C. Summers and M. J. Dinneen, "Improved mixed-example data augmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2019, pp. 1262–1270.

[27] A. F. M. S. Uddin, M. Sirazam Monira, W. Shin, T. Chung, and S.-H. Bae, "SaliencyMix: A saliency guided data augmentation strategy for better regularization," 2020, *arXiv:2006.01791*. [Online]. Available: http://arxiv.org/abs/2006.01791

[28] H. Wang, S. Ge, Z. Lipton, and E. P. Xing, "Learning robust global representations by penalizing local predictive power," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 10506–10518.

[29] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1492–1500.

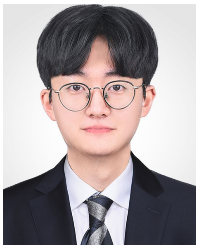
[30] S. Yun, D. Han, S. Chun, S. J. Oh, Y. Yoo, and J. Choe, "CutMix: Regularization strategy to train strong classifiers with localizable features," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6023–6032.

[31] S. Zagoruyko and N. Komodakis, "Wide residual networks," 2016, *arXiv:1605.07146*. [Online]. Available: http://arxiv.org/abs/1605.07146

[32] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," 2017, *arXiv:1710.09412*. [Online]. Available: http://arxiv.org/abs/1710.09412

[33] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *Proc. AAAI*, 2020, pp. 13001–13008.

[34] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2921–2929.



**YOUMIN KIM** received the bachelor's degree from the Department of Computer Science and Engineering, Kyung Hee University, South Korea, in 2019, where he is currently pursuing the M.S. degree. His research interests include data augmentation and knowledge distillation for deep neural networks.

**A. F. M. SHAHAB UDDIN** received the B.S. and M.S. degrees from the Department of Information and Communication Engineering, Islamic University, Kushtia, Bangladesh, in 2015 and 2016, respectively. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, Kyung Hee University, Suwon, South Korea. His research interests include general problems in machine learning, image quality assessment, perceptual image processing, and inverse problems in image processing.

**SUNG-HO BAE** (Member, IEEE) received the B.S. degree from Kyung Hee University, South Korea, in 2011, and the M.S. and Ph.D. degrees from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2012 and 2016, respectively. From 2016 to 2017, he was a Postdoctoral Associate with the Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology (MIT), MA, USA. Since 2017, he has been an Assistant Professor with the Department of Computer Science and Engineering, Kyung Hee University. He has been involved in model compression/interpretation for deep neural networks and inverse problems in image processing and computer vision.