IEICE TRANSACTIONS

on Information and Systems

DOI:10.1587/transinf.2024EDP7016

Publicized:2024/07/26

This advance publication article will be replaced by the finalized version after proofreading.

A PUBLICATION OF THE INFORMATION AND SYSTEMS SOCIETY

The Institute of Electronics, Information and Communication Engineers
Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

PAPER

Degraded image classification using knowledge distillation and robust data augmentations

Dinesh DAULTANI[†a)], *Nonmember*, Masayuki TANAKA[†], Masatoshi OKUTOMI[†], *Members*, *and* Kazuki ENDO[††], *Nonmember*

SUMMARY Image classification is a typical computer vision task widely used in practical applications. The images used for training image classification networks are often clean, i.e., without any image degradation. However, Convolutional neural networks trained on clean images perform poorly on degraded or corrupted images in the real world. In this study, we effectively utilize robust data augmentation (DA) with knowledge distillation to improve the classification performance of degraded images. We first categorize robust data augmentations into geometric-and-color and cut-and-delete DAs. Next, we evaluate the effectual positioning of cut-and-delete DA when we apply knowledge distillation. Moreover, we also experimentally demonstrate that combining the RandAugment and Random Erasing approach for geometric-and-color and cut-and-delete DA improves the generalization of the student network during the knowledge transfer for the classification of degraded images.

key words: *Classification, Image Degradation, Knowledge Distillation, Data Augmentation, Auto augmentation*

1. Introduction

Computer vision with machine learning techniques is vital in transportation [1], [2], health care [3], agriculture [4], [5], retail [6], manufacturing [7], and satellite imagery [8] applications. One such common computer vision task is image classification, where the machine learning model predicts the class of a given image. Image classification tasks are primarily performed based on Convolutional Neural Networks (CNN). To train CNN models, a wide variety of clean images are used without any image degradation. However, in the real world, images usually include some degradations, for example, (1) lossy image compressions such as JPEG and HEVC, (2) blur introduced due to out-of-focus camera or sudden movements during the image capture process, (3) noise artifacts due to low light or camera sensor limitations. Therefore, we aim to improve the classification performance of degraded images. In this study, we focus on four types of degradations, i.e., JPEG compression, salt-and-pepper noise (SAPN), Gaussian blur (Blur), and additive white Gaussian noise (AWGN).

As previously shown in our study [9], the performance of degraded image classification can be improved using knowledge distillation and data augmentation (DA) methods such as Cutout [10]. However, we did not explore other robust data augmentation methods in our previous study [9]. Hence, the primary purpose of our study is to explore several robust data augmentation methods with knowledge distillation for degraded image classification further. We first categorize data augmentations as geometric-and-color DA and cut-and-delete DA to further explore DAs for knowledge distillation.

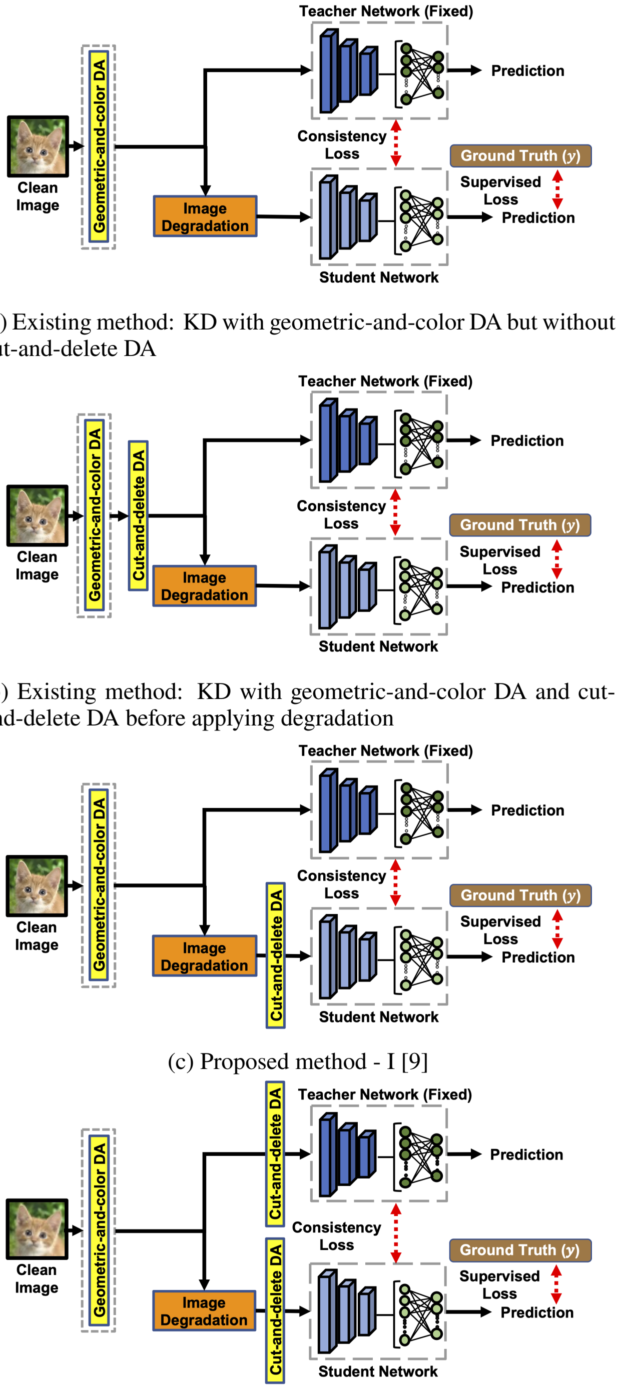
Images augmented through geometric-and-color DA still look like natural images; cut-and-delete DA augmented images help increase the network's performance, but their appearance is unnatural. Specifically, geometric-and-color DA includes geometric-based transforms such as crop, rotation, and affine, which alter the geometric attributes of an image, and color-based transforms such as brightness, contrast, and tone, which preserve the geometric properties of the image but alter the photometric aspects of the image. Our study classifies geometric-and-color DA as standard DA and auto augmentation approaches. In standard DA, we can apply either of the geometric-and-color DA transformations described above; conversely, auto augmentation approaches rely on automatically determining the optimal transformations for a given task and a predefined set of geometric-and-color transformations. Besides, cut-and-delete DA includes masking augmentations such as Cutout [10], which occludes/hides part of the image, subsequently forcing the models to focus on the whole image rather than a few essential aspects of the image, which leads to a better generalization.

Fig. 1 shows different cut-and-delete DA positions with knowledge distillation. Usually, we begin by applying geometric-and-color DA on the input image, followed by the application of cut-and-delete DA [10], [11]; hence, we fix the geometric-and-color DA position after the input image to ensure the desired sequence. Figs. 1 (a) and (b) are existing methods; on the other hand, (c) and (d) are our proposed methods. Figs. 1 (a) and (b) methods commonly apply DAs to clean images before image degradation while training a student network. Specifically, the method of Fig. 1 (a) includes only geometric-and-color DAs. Since cut-and-delete DA enhances the classification performance, the method depicted in Fig. 1 (b) includes a cut-and-delete DA following geometric-and-color DA. Next, Fig. 1 (c) is our proposed method - I [9] that applies cut-and-delete DA only to the degraded image generated by the image degradation module. At the end, Fig. 1 (d) represents our proposed method - II,

[†] The authors are with the Department of Systems and Control Engineering, Tokyo Institute of Technology, Tokyo, Japan

[††] The author is with the Department of Business, Teikyo Heisei University, Tokyo, Japan

a) E-mail: ddaultani@ok.sc.e.titech.ac.jp



(a) Existing method: KD with geometric-and-color DA but without cut-and-delete DA

(b) Existing method: KD with geometric-and-color DA and cut-and-delete DA before applying degradation

(c) Proposed method - I [9]

(d) Proposed method - II

Fig. 1: Several positioning of cut-and-delete DA with KD, where yellow, orange, and blue/green blocks represent data augmentation methods, image degradation, and teacher/student network backbones, respectively. Geometric-and-color DA includes auto augmentation as part of our proposed method - II.

which applies geometric-and-color DA to clean images and then cut-and-delete DA to both clean and degraded images, where geometric-and-color DA includes not only standard DA but also auto augmentation methods compared to the other three approaches. We also share the experimental results for comparing these four methods in the sections' 4.5 cut-and-delete DA experiments.

Our proposed method contributions are listed as follows: We demonstrate that applying cut-and-delete DA before the degradation operation in the data augmentation pipeline during the distillation reduces performance. Moreover, we investigate several variations of geometric-and-color DA and cut-and-delete DA approaches to demonstrate that the geometric-and-color DA and cut-and-delete DA help to improve the robustness and generalization for the classification of the degraded images. We use geometric-and-color DA to increase the variety of clean input images. At the same time, we apply the cut-and-delete DA on both clean and degraded images to achieve robust performance for degraded image classification. Specifically, we apply RandAugment [12] for geometric-and-color DA and Random Erasing [11] for cut-and-delete DA in our proposed method II. Furthermore, we provide empirical results comparing our proposed methods I and II with previous works on several datasets such as CIFAR-10, CIFAR-100, and Tiny ImageNet and subsequently with several degradations such as JPEG, SAPN, Blur, and AWGN.

This study is an extended version of our previous conference paper; the main differences from our proposed method - I [9] are as follows: (1) We evaluate the optimal positioning of cut-and-delete DA while keeping positioning of degradation and geometric-and-color DA constant. (2) We further explore robust geometric-and-color DA methods such as auto augmentation methods and other cut-and-delete data augmentation methods.

2. Related Works

Degraded image classification: There are several approaches for degraded image classification. In a sequential network with a restoration network [13] or an enhancement network [14], one typically needs to reconstruct/enhance the image using either a convolutional network or specific filters employed in the degradation model. However, restoring or enhancing the image without knowing the prior degradation is challenging, and it increases the number of parameters in the network. Moreover, Pei *et al.* [15] have shown that restored images do not improve the performance of CNN-based methods for classifying degraded images compared to training the model directly on degraded images. Alternatively, by utilizing a limited number of images, Deep Degradation Prior (DDP) [16] improves the classification performance of degradation, such as fog, contrast, and brightness, by shortening the feature mismatch between clean and degraded images. However, our focus is not to utilize limited images but all the images available in the simulated dataset and concentrate on more commonly occurring degradations.

The following studies have explored different architectures with several consistency losses, vector quantization, self-attention, the estimator of degradation parameters, restoration networks, scale-estimators / bias-estimators, and ensembles. Pei *et al.* [17] proposed a consistency-guided network based on knowledge distillation with category consistency, visual attention alignment, and semantic consistency losses to classify degraded images. Next, Yang *et al.* [18] introduced vector quantization for low-quality image recognition that includes codebook modules and self-attention invariant to image quality, although making the proposed method more than twice in number of parameters. Endo *et al.* [19] focused on creating a network ensemble based on a network trained with clean images and another network trained with restored images. Next, the feature adjustor method [20] contains two feature extractors, scale/bias estimators and degradation levels estimators, outperforming the performance of Pei *et al.* [17]. However, the feature adjustor makes the computational parameters more than twice the ones of the typical feature extractor. Additionally, during the training process, the feature adjustor requires values of degradation levels that might not be available for real-world images. Our proposed method allows the classification network not to need additional sub-networks like an estimator of degradation levels seen in Endo *et al.* [19], [20] or vector quantization modules seen in Yang *et al.* [18].

Data augmentation: Data augmentation (DA) is a general technique used to increase the diversity of the training data and robustness of the network. To increase the diversity of the training data, we apply geometric and color-based transforms such as random flip and brightness, respectively. There are various methods to perform geometric and color-based transforms [21], making it challenging to search for the best combination among different DA method candidates. Auto augmentation approaches, such as AutoAug [22] and RandAugment [12], assist us in finding the best combination of geometric and color-based transforms automatically. AutoAug finds the best policy of data transformation sequences for a specific dataset given a fixed neural network architecture using a search method based on reinforcement learning. We apply the relevant policies, described in AutoAug [22], to the classification of degraded images for the CIFAR and Tiny Imagenet datasets [23]. Similarly, RandAugment applies a DA operation to each image while sampling from a set of image transformations.

On the other hand, we erase part of the image as a regularization method to increase the robustness of the network. In this paper, we call this type of DA cut-and-delete DA. Cut-and-delete DA methods have also improved the performance of computer vision tasks, where we erase patches from the images and replace them with specific values [24]. Erased patches work as an occlusion for objects in an image, increasing the generalization performance of convolutional neural networks [10]. In cutout [10], a square size region is removed and replaced by mean values of the dataset. Next, In Random Erasing [11], a patch is randomly erased with a given region upper and lower limit and aspect ratio. This study explores AutoAug and RandAugment for geometric-and-color DAs; similarly, we explore cutout and Random Erasing for cut-and-delete DAs.

Knowledge distillation with data augmentation: Several studies have assessed the efficacy of data augmentation with the knowledge distillation [25] framework. Wang *et al.* [26] have proposed a KD-based approach that includes the input of images to the teacher/student network from both standard DA (random crop and flip) and stronger DA (CutMix [27]), leading to twice the input sample size. The losses for Standard DA's input images are calculated based on Kullback-Leibler Divergence (KLD) and cross-entropy (CE) loss. On the other hand, the loss for stronger DA input images is only KLD since CutMix [27] assigns a linearly interpolated label to an augmented sample, which might be different from the ground truth. However, our proposed method has significant differences specific to data augmentation. Among others, some of the differences include (1) Stronger DA includes a combination of auto augmentation and cut-and-delete DA in our proposed method - II rather than CutMix applied in Wang *et al.* [26] with an entropy-based approach for selection of images, (2) We do not directly feed the standard DA images to either the teacher/student network and (3) Classification of degraded images is our target rather than typical clean images.

3. Proposed Method

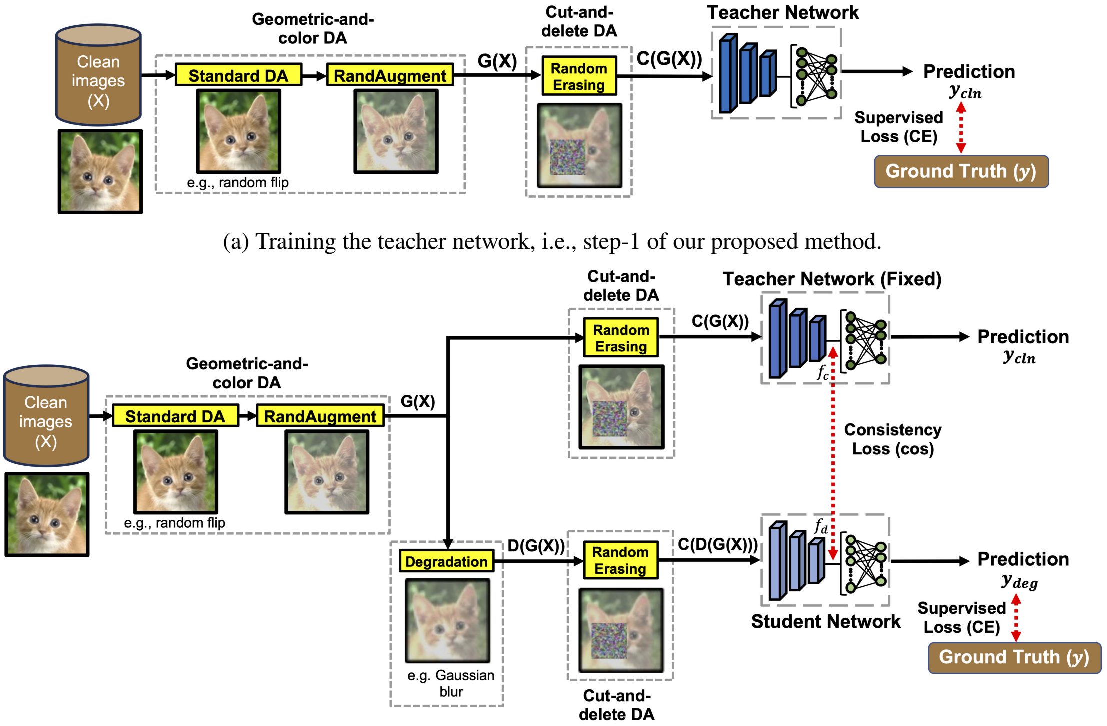
Our proposed method comprises three steps: First, we train a teacher network where we apply geometric-and-color DA and cut-and-delete DA on the clean images as shown in Fig. 2a. Next, we train a target student network where we apply geometric-and-color DA and cut-and-delete DA along with knowledge transfer from the pre-trained teacher network as shown in Fig. 2b. Lastly, we perform the inference of degraded images on the target student network without any data augmentations for the evaluation. More details about the first and second steps have been described below.

3.1 Training teacher network

As shown in Fig. 2a, the clean images are input first to geometric-and-color DA, resulting in $\boldsymbol{G}(\boldsymbol{X})$, where $\boldsymbol{X}$ is input clean images and $\boldsymbol{G}$ represents the operator of geometric-and-color DA. Then, we input images $\boldsymbol{G}(\boldsymbol{X})$ to cut-and-delete DA, i.e., Random Erasing [11]. Geometric-and-color DA and cut-and-delete DA guide the teacher network to improve generalization performance on clean images. Next, the augmented images $\boldsymbol{C}(\boldsymbol{G}(\boldsymbol{X}))$ are input to the teacher network, where $\boldsymbol{C}$ denotes the operator of cut-and-delete DA. Subsequently, we train the teacher network using the cross-entropy loss function against the ground truth labels of the clean image.

3.2 Training student network

On the other hand, as shown in Fig. 2b, we transfer the knowledge from a pre-trained teacher network to the target student network so that the features of the student network trained on degraded images are consistent with the features of the



(a) Training the teacher network, i.e., step-1 of our proposed method.

(b) Training the student network using knowledge distillation, i.e., step-2 of our proposed method.

Fig. 2: Illustration of our proposed method - II. The yellow and brown color blocks represent DA methods and datasets, respectively. Teacher and student networks are represented in blue/green blocks, whereas blue and green color blocks represent CNN layers and the classifier.

teacher network trained on clean images. First, we apply several geometric-and-color DA methods, such as standard DA and RandAugment [12], on the clean images, resulting in $\boldsymbol{G}(\boldsymbol{X})$ as shown in Fig. 2b. Next, we segregate the $\boldsymbol{G}(\boldsymbol{X})$ images into the teacher and student networks, where we apply image degradations, such as Gaussian blur, to the inputs of the student network. At the same time, we apply cut-and-delete DA to the inputs of both teacher and student networks. At last, we pass the augmented image denoted by $\boldsymbol{C}(\boldsymbol{G}(\boldsymbol{X}))$ to the teacher network and degraded augmented image denoted by $\boldsymbol{C}(\boldsymbol{D}(\boldsymbol{G}(\boldsymbol{X})))$ to the student network, where consistency loss (CL) is calculated based on intermediate features f_c and f_d. Moreover, we apply cross-entropy (CE) loss between the ground truth and predictions.

The loss function of the target student network training in our proposed approach includes supervised and consistency loss. The total loss function L for the distillation process is defined as follows,

$$L = \gamma_{sup} L_{sup} + \gamma_{con} L_{con}\,, \tag{1}$$

$$L_{sup}(y, y_{deg}) = -\,y \log(y_{deg}), \tag{2}$$

$$L_{con}(f_c, f_d) = 1 - \frac{f_c \cdot f_d}{\|f_c\|\,\|f_d\|}. \tag{3}$$

where L_{sup} and L_{con} represent supervised loss and consistency loss functions, respectively. Consequently, γ_{sup} and γ_{con} represent the weights of supervised and consistency loss functions. The supervised loss function L_{sup}, i.e., cross-entropy loss, is evaluated between student network's prediction y_{deg} and ground truth y as shown in equation 2. We evaluate the consistency loss function L_{con} between intermediate outputs f_d and f_c as shown in equation 3. Specifically, we utilize the cosine similarity (cos) loss to transfer information from the teacher network to the student network as shown in the equation 3 similar to Endo *et al.* [20]. Additionally, we noticed that single-layer KD applied between the convolutional layers and classifier works best compared to other possible positions and multi-layer feature-based KD.

Moreover, in our proposal for both teacher and student network training, we specifically opt for the RandAugment [12] approach of auto augmentation for geometric-and-color DA and Random Erasing for cut-and-delete DA as it performs better than the other methods. Later in section 4.5, we explain more details about comparing several geometric-and-color and cut-and-delete DA methods.

4. Experiments

4.1 Training procedure

Since our proposed methods rely on knowledge distillation, we use two experimental configurations for training, i.e., teacher and student network training. We use an SGD optimizer with an initial learning rate of 0.1, a momentum factor of 0.9, and an L2 penalty weight decay of 0.0005 for teacher network training. The teacher network has been trained for 200 epochs with a multi-step learning rate scheduler with a decrease in learning rate at 60, 120, and 160 epochs by a multiplicative factor of 0.2. Conversely, we initialize weights with teacher models trained on clean images for student network training since the architecture is the same. For student network training, we use the RAdam optimizer with an initial learning rate of 0.001 and an L2 penalty weight decay of 0.0001. The student network has trained for 100 epochs with a Cosine Annealing learning rate scheduler. All experiments use PyTorch [28] library version 1.12.

4.2 Backbones

Focus of this study is to study the effectiveness of data augmentation and knowledge distillation on the classification of degraded images. Hence, to exhibit the effectiveness of our proposed methods, we utilize two commonly used classification backbones, i.e., ResNet and ShakePyramidNet. Specifically, we used the acronyms BackboneX-Y to represent student networks, where X and Y represent the number of layers in the teacher and student networks, respectively. For example, ResNet56-56 represents the teacher backbone of ResNet56 [29] with the student backbone of ResNet56. Similarly, ShakePyramidNet110-110 (or SPN110-110 in short) represents the teacher and student network of PyramidNet [30] with shake drop regularization [31] with 110 layers and alpha of 270.

4.3 Datasets and processing

Primarily, we have evaluated our proposed methods on CIFAR-10 and CIFAR-100 datasets to compare with existing methods. Furthermore, we have evaluated our proposed methods on the Tiny ImageNet dataset. Tiny ImageNet contains 100k images for training with 64 × 64 resolution of 200 classes, which is comparatively larger than the datasets tested in previous works [9], [19], [20], [32], i.e., CIFAR-10 / CIFAR-100. All experiments include preprocessing with data augmentation of random horizontal flips and random crops as a part of geometric-and-color DA.

In addition, the proposed architecture in Fig. 2 illustrates further preprocessing with different data augmentation, such as geometric-and-color DA and cut-and-delete DA. Specifically, we apply RandAugment similar to the implementation by their authors [12], where transform operations include geometric and color transforms such as Identity, Rotate, ShearX, ShearY, TranslateX, TranslateY, AutoContrast, Equalize, Solarize, Color, Posterize, Contrast, Brightness, and Sharpness. In addition, there are two hyperparameters with the RandAugment approach, i.e., the number of transformation operations to apply (N) and the magnitude for the transformations (M). For CIFAR-10/CIFAR-100 dataset $N = 1$ and $M = 5$ and for Tiny ImageNet dataset, $N = 2$ and $M = 9$.

Proposed method - I requires tuning for cutout length for specific datasets or tasks. However, our proposed method - II uses Random Erasing for cut-and-delete DA, which does not require hyperparameter tuning specific to a given task and dataset. Specifically, the hyperparameter values include the probability that a random erasing operation will be applied (p = 0.5), the range for a proportion of erased image as compared with the original size of the image (lower bound = 0.02, upper bound = 0.4), and aspect ratio of the erased image (0.3, 3.3). Consequently, we use the same hyperparameters of Random Erasing with all our experiments on CIFAR-10, CIFAR-100, and Tiny ImageNet datasets in the subsequent sections.

4.4 Evaluation metrics

The networks used for image classification tasks are primarily measured using an accuracy metric. Since we need to measure the accuracy at each degradation level, we have used the interval mean accuracy (IMA) metric similar to Endo *et al.* [19], [20], [32], [33] for measuring the accuracy at different degradation levels. The equation for calculating interval mean accuracy is defined below in equation 4. IMA for the given model parameter θ, and degradation levels Q_l, Q_u is represented by $\overline{Acc}$. $\mathbf{X}$ represents clean input images for respective $\mathbf{Y}$ ground truth labels where the degradation level q varies between Q_l and Q_u, and D denotes the image degradation operator.

$$\overline{Acc}(\theta, Q_l, Q_u) = \frac{\sum_{q=Q_l}^{Q_u} Acc(f(D(\mathbf{X}, q) : \theta), \mathbf{Y})}{Q_u - Q_l + 1} \quad (4)$$

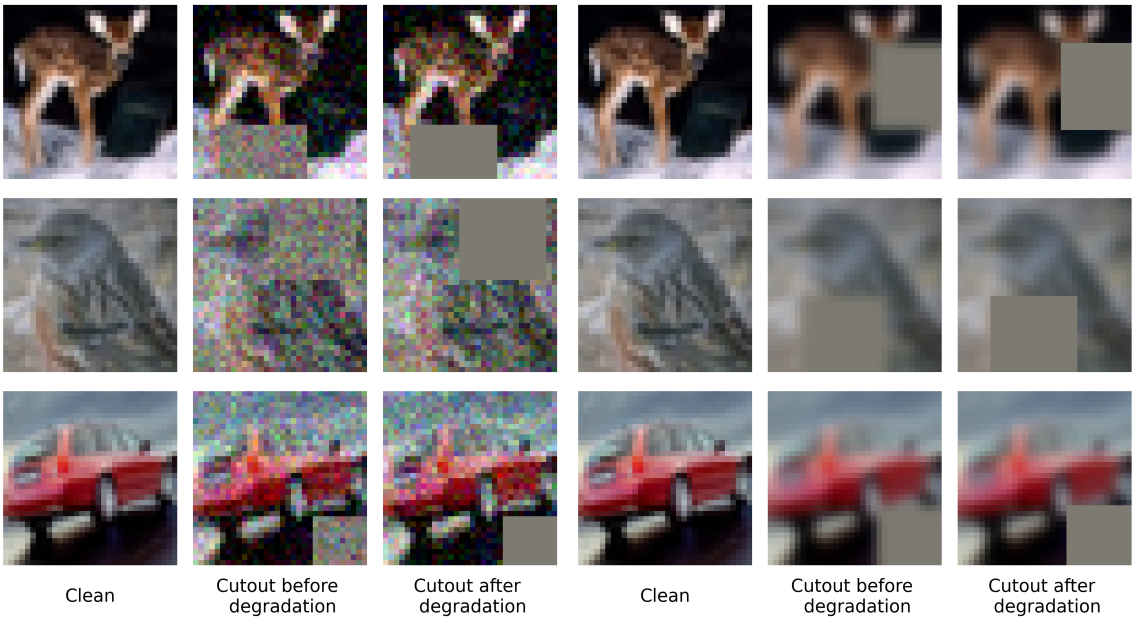
4.5 Cut-and-delete DA positions and different variations of data augmentations

We investigate the data augmentation setup for knowledge distillation. First, we perform experiments on different cut-and-delete DA positions. Then, we perform experiments combining several cut-and-delete DA with geometric-and-color DA method variations.

We conducted several experiments for different positions of cut-and-delete DA shown in Fig. 1 to improve the robustness and performance of student networks. We apply standard DA for geometric-and-color DA and cutout [10] for cut-and-delete DA for training student networks, as this is a simple setting to determine the effective position for cut-and-delete DA. Additionally, we used a common teacher network trained with standard DA for geometric-and-color DA and cutout [10] for cut-and-delete DA for fair comparisons. Table 1 shows the average accuracies of four positions to apply cut-and-delete DA as shown in Fig. 1 on the CIFAR-10 dataset with ResNet56-56 backbone. Figs. 1 (b), (c), and (d) include cut-and-delete DA. We apply cut-and-delete DA before image degradation in Fig. 1 (b). In contrast, we apply cut-and-delete DA after image degradation in Figs. 1 (c) and (d). From Table 1, we can find cut-and-delete DA positions

Table 1: Experiments with different cut-and-delete DA positions for image classification of degraded images on the CIFAR-10 dataset. † represents cut-and-delete DA applied after the geometric-and-color DA directly to the clean images and before the degradation module. On the other hand, ‡ represents cut-and-delete DA is applied just before feeding the images to the teacher or student network. The symbols' absence indicates that no cut-and-delete DA is applied on respective images. Experiment results are based on three runs with different random seeds.

Cut-and-delete			$\overline{Acc}$(All)				
Position	Clean Image	Degrade Image	JPEG	SAPN	Blur	AWGN	AVG
Fig. 1a			0.874 ± 0.0000	0.943 ± 0.0005	0.838 ± 0.0005	0.893 ± 0.0005	0.8870
Fig. 1b	†	†	0.879 ± 0.0009	0.942 ± 0.0012	**0.847** ± 0.0012	0.897 ± 0.0012	0.8913
Fig. 1c		‡	**0.882** ± 0.0005	0.946 ± 0.0005	0.845 ± 0.0000	**0.901** ± 0.0014	0.8935
Fig. 1d	‡	‡	0.881 ± 0.0008	**0.948** ± 0.0008	0.846 ± 0.0005	0.900 ± 0.0012	**0.8938**



(a) AWGN with degradation level = 20.

(b) Gaussian blur with degradation level = 1.

Fig. 3: Sample images from CIFAR-10 dataset while applying cutout augmentation before and after degradation as described in Fig. 1 (b), (c), and (d) architectures.

of Figs. 1 (c) and (d) are slightly better than those of Figs. 1 (a) and (b).

To analyze the differences between Fig. 1 (b), (c), and (d) architectures, we provide sample images in Fig. 3 for AWGN and Gaussian blur degradations. We apply cutout for cut-and-delete DA to visualize the independent impact on different architectures. As shown in Fig. 3 (a), when cutout augmentation is applied before degradation, the added degradation can introduce noise in the cutout region instead of constant values when the cutout is applied after degradation. Contrarily, it is visually challenging to discern the differences between the cutout before/after degradation in Fig. 3 (b) for Gaussian blur. Although the images appear identical with pixel values as grey, the pixel values modestly diverge when degradation is applied following cutout, making the changes not visually apparent in Fig. 3 (b). Moreover, we exhibit that applying cut-and-delete DA during the distillation process before the degradation operation in the data augmentation pipeline somewhat changes the network's capacity to learn the features and inherently leads to relatively slightly lower performance.

Table 1 shows the performances of the cut-and-delete DA positions of Fig. 1 (c) and (d) are comparable. Nonetheless, we would like to standardize the position of cut-and-delete DA over different degradations; hence, we consider the average performance over all degradations. Therefore, based on the average (AVG) results in Table 1, we select Fig. 1 (d) as our proposed cut-and-delete DA position in the proposed method - II. Furthermore, since the standard deviation in Table 1 is very small, we apply only one random seed for further experiments.

Based on Fig. 1 (d) cut-and-delete DA position, we apply several variations of geometric-and-color and cut-and-delete DA methods to compare DA methods for degraded image classification. We apply the same data augmentation methods for training the teacher network as the combination applied to the student network. First, we apply different DA variations on the CIFAR-10 dataset with four different

Table 2: Experiments with several geometric-and-color DA and cut-and-delete DA approaches for the clean and degraded image classification.

(a) CIFAR-10 dataset with ResNet56-56 backbone

Geometric-and-color DA	Cut-and-delete DA	Performance: $\overline{Acc}$(All)				
		JPEG	SAPN	Blur	AWGN	AVG
Standard DA	-	0.864	0.932	0.829	0.883	0.8770
Standard DA	Cutout [10]	0.881	0.947	0.846	0.901	0.8938
Standard DA	Random Erasing [11]	0.880	0.946	0.846	0.901	0.8933
Standard DA + AutoAug [22]	Cutout [10]	**0.888**	**0.954**	0.849	0.907	**0.8995**
Standard DA + AutoAug [22]	Random Erasing [11]	0.886	0.948	0.850	0.903	0.8968
Standard DA + RandAugment [12]	Cutout [10]	0.885	0.951	**0.853**	0.907	0.8990
Standard DA + RandAugment [12]	Random Erasing [11]	0.886	0.951	0.851	**0.908**	0.8990

(b) CIFAR-100 dataset with SPN110-110 backbone

Geometric-and-color DA	Cut-and-delete DA	Performance: $\overline{Acc}$(All)				
		JPEG	SAPN	Blur	AWGN	AVG
Standard DA	-	0.716	0.835	0.691	0.748	0.7475
Standard DA	Cutout [10]	0.729	0.854	0.690	0.757	0.7575
Standard DA	Random Erasing [11]	0.729	0.851	0.692	0.758	0.7575
Standard DA + AutoAug [22]	Cutout [10]	0.728	0.855	0.687	0.755	0.7563
Standard DA + AutoAug [22]	Random Erasing [11]	**0.734**	**0.860**	0.692	0.758	0.7610
Standard DA + RandAugment [12]	Cutout [10]	**0.734**	0.859	0.693	0.757	0.7608
Standard DA + RandAugment [12]	Random Erasing [11]	0.731	0.856	**0.697**	**0.761**	**0.7613**

degradation types using ResNet56-56 backbones as shown in Table 2 (a). The first row shows the baseline when we apply only standard DA, i.e., random crop and flip as geometric-and-color DA without any cut-and-delete DA. We use three types of geometric-and-color DA: standard DA only, AutoAug [22] with standard DA, and RandAugment [12] with standard DA. For cut-and-delete DA, we apply cutout [10] and Random Erasing [11] methods.

Table 2 (a) shows the effectiveness of several cut-and-delete and geometric-and-color DA method variations for classifying degraded images. Since several approaches performance is close to each other, we conduct further comparisons of all combinations with ShakePyramidNet backbone [30], [31] on the CIFAR-100 dataset with four different degradation types as shown in Table 2 (b). Those results demonstrate that AutoAug [22] + Random Erasing [11], RandAugment [12] + cutout [10], and RandAugment [12] + Random Erasing [11] have comparable performance. To standardize data augmentation methods for cut-and-delete DA and geometric-and-color DA, we choose RandAugment [12] + Random Erasing [11] methods for our proposed method - II based on average performance over all degradations from Table 2 (b). Moreover, we utilize different types of cut-and-delete and geometric-and-color DA to provide a general framework on how these methods can be applied to train neural network for the classification of degraded images.

4.6 Comparison with existing methods

Table 3 summarizes the differences between existing and proposed methods regarding model architecture, data augmentation, and model training. ”Clean” and ”DEG” methods are straightforward methods to train the model with clean and degraded images using cross-entropy loss functions and standard DA methods such as random crop and random horizontal flip. The ”DIST” method is a typical knowledge distillation method in which the student network is trained from a pre-trained teacher network using the KLD loss function after the softmax function.

The Feature Adjustor ”FA” method [20] uses a consistency loss function, i.e., cosine similarity, and an MSE loss function between actual and predicted degradation levels. Next, our proposed method, i.e., ”Ours - I,” is based on knowledge distillation; however, the consistency loss function is applied after the convolution blocks and before the last average pooling layer of the network, where the performance of the network is improved using cutout data augmentation [9]. Lastly, the ”Ours - II” method represents our proposed method based on knowledge distillation and more robust data augmentation methods such as RandAugment and Random Erasing cut-and-delete augmentation. In the subsequent sections, we have compared our proposed methods with existing methods on CIFAR-10/CIFAR-100 datasets for image degradation of JPEG compression, AWGN, Gaussian blur, and salt-and-pepper noise. In addition, we used the Tiny ImageNet dataset to show a more realistic comparison of a larger dataset in section 4.6.2.

4.6.1 Comparison on CIFAR-10 and CIFAR-100 datasets

Table 4 shows the comparisons on CIFAR-10 and CIFAR-100 datasets with degradation of JPEG compression. Given the effectiveness of the ”FA” method on clean images [20], the performance of the ”FA” method is better or similar to the ”Clean” method on the ”Clean Image” degradation interval. On next lower degradation interval, i.e., $\overline{Acc}$(1-20) performance of ”Ours - I”, ”Ours - II” and ”FA” are almost similar on both CIFAR-10 dataset and CIFAR-100 dataset. On other degradation intervals, i.e., $\overline{Acc}$(21-40), $\overline{Acc}$(41-60), $\overline{Acc}$(81-100); performance of ”Ours - II” is

Table 3: Architecture differences between previous existing methods and our proposed approaches.

	Existing methods				Proposed	
	Clean	DEG	DIST	FA [20]	Ours - I [9]	Ours - II
Loss functions	CE	CE	CE+CL	CL+DL	CE+CL	CE+CL
CL function	-	-	KLD	cos	cos	cos
CL location	-	-	after softmax	after avg pool	after conv blocks	after conv blocks
Training image	clean	degrade	clean & degrade	clean & degrade	clean & degrade	clean & degrade
Geometric-and-color DA	standard DA	standard DA	standard DA	standard DA	standard DA	standard DA + RandAugment [12]
Cut-and-delete DA	-	-	-	-	cutout [10]	Random Erasing [11]

better than other existing methods; except the ”DEG” method on $\overline{Acc}$(81-100) degradation interval. Overall, ”Ours - II” performs consistently well on all degradation levels except the very low degradation levels as shown in Table 4 (b), showing the effectiveness of our proposed method for JPEG compressed image classification.

Table 5 compares our proposed methods with existing methods on salt-and-pepper noise degradation on CIFAR-10 and CIFAR-100 datasets. Overall, our proposed method - II performs best on all degradation intervals, including the clean images. Additionally, our proposed method, Ours - I, achieves slightly lower performance but is almost comparable. Due to the effective data augmentations methods in our proposed methods, both of our proposed approaches achieve overall better $\overline{Acc}$(All) performance than FA method, i.e., ”Ours - II” with 0.973 and Ours - I with 0.970 as compared to FA with 0.963 on CIFAR-10 dataset. Similarly, our proposed approaches perform better than the FA method on CIFAR-100 datasets, i.e., ”Ours - II” with 0.856, Ours - I with 0.852, and FA with 0.823.

Table 6 compares our proposed methods with previous methods for the Gaussian blur degradation method on CIFAR-10 and CIFAR-100 datasets. Similar to JPEG compression, for lower degradation intervals, i.e., ”Clean Image” and $\overline{Acc}$(0.1-1), the ”FA” and ”Clean” methods can perform well. Specifically, for the ”Clean Image” degradation level, FA performs the best on the CIFAR-10 dataset, i.e., 0.966, and the ”Clean” method performs the best on the CIFAR-100 dataset, i.e., 0.841. For $\overline{Acc}$(0.1-1) degradation interval, there is a tie between our proposed method ”Ours - I” and ”FA” on the CIFAR-10 dataset for IMA of 0.949. However, the best performance is with ”Ours - II”, i.e., 0.951. For the CIFAR-100 dataset, ”FA” performs the best, i.e., 0.805, and our proposed method - II is second best, i.e., 0.794. For higher degradation levels, $\overline{Acc}$(1.1-2), $\overline{Acc}$(2.1-3), $\overline{Acc}$(3.1-4), and $\overline{Acc}$(4.1-5), our proposed method ”Ours - II” performs the best for CIFAR-100 dataset with IMA of 0.760, 0.704, 0.640, and 0.577 respectively. For CIFAR-10 dataset, our proposed method ”Ours - II” outperforms the other methods on $\overline{Acc}$(1.1-2), $\overline{Acc}$(2.1-3) degradation intervals with IMA of 0.929 and 0.889. For higher degradation intervals $\overline{Acc}$(3.1-4) and $\overline{Acc}$(4.1-5) on

Table 4: Interval mean accuracy for the feature extractor based on ShakePyramidNet and Endo *et al.* [20] with degradation of JPEG compression on CIFAR-10 and CIFAR-100 datasets. Degradation levels are defined as $100 - JPEG\ quality\ factors$ where $JPEG\ quality\ factors$ range from 0 to 100 with a step size of 1.

(a) CIFAR-10 Dataset

Degradation Interval	Clean	DEG	DIST	FA	Ours - I	Ours - II
Clean Image	0.964	0.936	0.946	**0.966**	0.954	0.954
$\overline{Acc}$(1-20)	0.932	0.934	0.943	0.950	0.951	**0.952**
$\overline{Acc}$(21-40)	0.852	0.928	0.935	0.931	**0.941**	**0.941**
$\overline{Acc}$(41-60)	0.780	0.919	0.925	0.919	**0.931**	**0.931**
$\overline{Acc}$(61-80)	0.674	0.906	0.908	0.903	0.913	**0.915**
$\overline{Acc}$(81-100)	0.391	**0.806**	0.797	0.803	0.800	0.804
$\overline{Acc}$(All)	0.728	0.899	0.902	0.902	0.908	**0.909**

(b) CIFAR-100 Dataset

Degradation Interval	Clean	DEG	DIST	FA	Ours - I	Ours - II
Clean Image	**0.841**	0.765	0.788	0.836	0.798	0.799
$\overline{Acc}$(1-20)	0.747	0.762	0.783	**0.798**	0.794	0.794
$\overline{Acc}$(21-40)	0.605	0.750	0.770	0.762	0.775	**0.778**
$\overline{Acc}$(41-60)	0.512	0.738	0.754	0.739	0.756	**0.761**
$\overline{Acc}$(61-80)	0.389	0.718	0.729	0.711	0.731	**0.737**
$\overline{Acc}$(81-100)	0.144	0.575	0.574	0.565	0.578	**0.583**
$\overline{Acc}$(All)	0.483	0.709	0.723	0.716	0.727	**0.731**

Table 5: Interval mean accuracy for the feature extractor based on ShakePyramidNet and Endo *et al.* [20] with degradation of salt-and-pepper noise on CIFAR-10 and CIFAR-100 datasets. Degradation levels represent the noise density ranging from 0.00 to 0.25 with a step size of 0.01, where 0.00 means a clean image.

(a) CIFAR-10 Dataset

Degradation Interval	Clean	DEG	DIST	FA	Ours - I	Ours - II
Clean Image	0.964	0.961	0.964	0.967	0.972	**0.974**
$\overline{Acc}$(0.01-05)	0.619	0.961	0.963	0.965	0.972	**0.973**
$\overline{Acc}$(0.06-0.1)	0.255	0.959	0.962	0.965	0.971	**0.973**
$\overline{Acc}$(0.11-0.15)	0.129	0.958	0.960	0.963	0.970	**0.973**
$\overline{Acc}$(0.16-0.2)	0.105	0.958	0.959	0.961	0.970	**0.972**
$\overline{Acc}$(0.21-0.25)	0.101	0.956	0.957	0.959	0.968	**0.972**
$\overline{Acc}$(All)	0.270	0.958	0.960	0.963	0.970	**0.973**

(b) CIFAR-100 Dataset

Degradation Interval	Clean	DEG	DIST	FA	Ours - I	Ours - II
Clean Image	0.841	0.835	0.844	0.838	0.853	**0.859**
$\overline{Acc}$(0.01-05)	0.305	0.836	0.842	0.826	0.854	**0.858**
$\overline{Acc}$(0.06-0.1)	0.043	0.837	0.840	0.823	0.852	**0.858**
$\overline{Acc}$(0.11-0.15)	0.023	0.836	0.837	0.822	0.853	**0.857**
$\overline{Acc}$(0.16-0.2)	0.020	0.833	0.835	0.820	0.851	**0.854**
$\overline{Acc}$(0.21-0.25)	0.020	0.830	0.831	0.819	0.848	**0.853**
$\overline{Acc}$(All)	0.111	0.834	0.837	0.823	0.852	**0.856**

Table 6: Interval mean accuracy for the feature extractor based on ShakePyramidNet and Endo *et al.* [20] with degradation of Gaussian blur on CIFAR-10 and CIFAR-100 datasets. Degradation levels represent the standard deviation of the Gaussian blur filter ranging from 0.0 to 5.0 with a step size of 0.1, where 0.0 means clean image.

(a) CIFAR-10 Dataset

Degradation Interval	Clean	DEG	DIST	FA	Ours - I	Ours - II
Clean Image	0.964	0.930	0.949	**0.966**	0.951	0.953
$\overline{Acc}$(0.1-1)	0.860	0.928	0.945	0.949	0.949	**0.951**
$\overline{Acc}$(1.1-2)	0.237	0.910	0.919	0.911	0.928	**0.929**
$\overline{Acc}$(2.1-3)	0.174	0.885	0.880	0.881	0.886	**0.889**
$\overline{Acc}$(3.1-4)	0.170	**0.850**	0.834	0.838	0.833	0.837
$\overline{Acc}$(4.1-5)	0.171	**0.808**	0.783	0.789	0.783	0.785
$\overline{Acc}$(All)	0.335	0.877	0.874	0.875	0.877	**0.880**

(b) CIFAR-100 Dataset

Degradation Interval	Clean	DEG	DIST	FA	Ours - I	Ours - II
Clean Image	**0.841**	0.731	0.795	0.839	0.791	0.799
$\overline{Acc}$(0.1-1)	0.672	0.730	0.791	**0.805**	0.789	0.794
$\overline{Acc}$(1.1-2)	0.099	0.710	0.755	0.730	0.753	**0.760**
$\overline{Acc}$(2.1-3)	0.019	0.676	0.699	0.676	0.696	**0.704**
$\overline{Acc}$(3.1-4)	0.012	0.631	0.636	0.617	0.634	**0.640**
$\overline{Acc}$(4.1-5)	0.011	**0.575**	0.574	0.557	0.572	**0.577**
$\overline{Acc}$(All)	0.176	0.666	0.693	0.680	0.691	**0.697**

Table 7: Interval mean accuracy for the feature extractor based on ShakePyramidNet and Endo *et al.* [20] with degradation of AWGN on CIFAR-10 and CIFAR-100 datasets. Degradation levels represent the standard deviation of the Gaussian distribution ranging from 0 to 50 with a step size of 1, where a standard deviation of 0 means a clean image.

(a) CIFAR-10 Dataset

Degradation Interval	Clean	DEG	DIST	FA	Ours - I	Ours - II
Clean Image	0.964	0.946	0.953	**0.967**	0.959	0.965
$\overline{Acc}$(1-10)	0.876	0.945	0.951	0.956	0.958	**0.963**
$\overline{Acc}$(11-20)	0.536	0.936	0.941	0.937	0.945	**0.953**
$\overline{Acc}$(21-30)	0.208	0.924	0.923	0.922	0.928	**0.937**
$\overline{Acc}$(31-40)	0.130	0.904	0.897	0.903	0.905	**0.917**
$\overline{Acc}$(41-50)	0.109	0.865	0.850	0.881	0.860	**0.890**
$\overline{Acc}$(All)	0.383	0.915	0.913	0.921	0.920	**0.933**

(b) CIFAR-100 Dataset

Degradation Interval	Clean	DEG	DIST	FA	Ours - I	Ours - II
Clean Image	**0.841**	0.787	0.808	0.839	0.816	0.825
$\overline{Acc}$(1-10)	0.588	0.782	0.803	0.812	0.809	**0.818**
$\overline{Acc}$(11-20)	0.169	0.762	0.782	0.773	0.786	**0.794**
$\overline{Acc}$(21-30)	0.040	0.736	0.751	0.738	0.754	**0.764**
$\overline{Acc}$(31-40)	0.020	0.700	0.705	0.703	0.711	**0.730**
$\overline{Acc}$(41-50)	0.015	0.640	0.645	0.667	0.652	**0.694**
$\overline{Acc}$(All)	0.180	0.725	0.739	0.740	0.744	**0.761**

the CIFAR-10 dataset, the ”DEG” method outperforms the other methods. Overall, at $\overline{Acc}$(All) degradation interval, our proposed methods ”Ours - I”/”Ours - II” outperforms all other methods, including ”FA”.

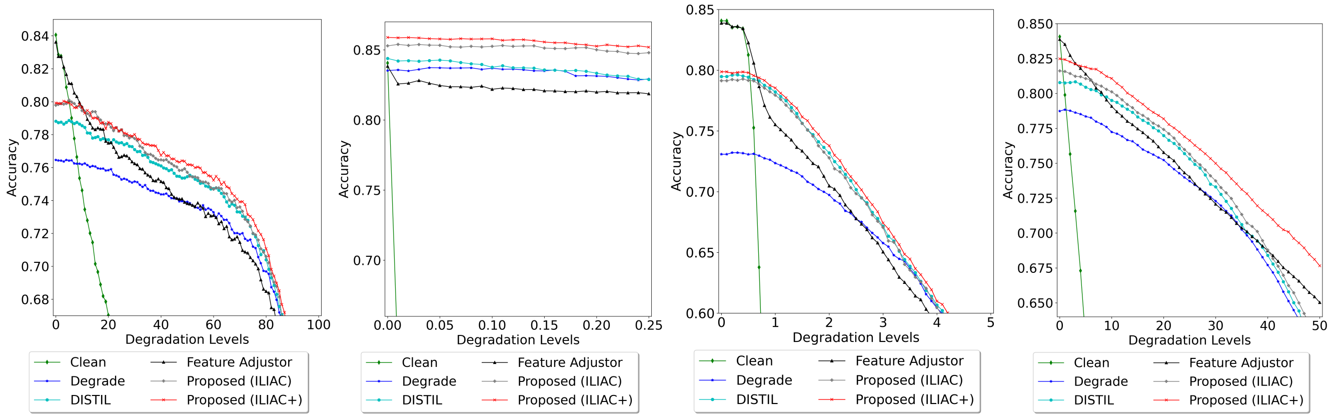
Table 7 compares our proposed methods with previous methods for AWGN degradation on CIFAR-10 and CIFAR-100 datasets. Similar to JPEG and Gaussian blur, the ”FA” and ”Clean” methods perform well for ”Clean Image” degradation intervals. Distinctively for the ”Clean Image” degradation level, ”FA” performs the best with an IMA of 0.967 on the CIFAR-10 dataset, and the ”Clean” method performs the best on the CIFAR-100 dataset with an IMA of 0.841. For the remaining five degradation intervals, i.e., $\overline{Acc}$(1-10), $\overline{Acc}$(11-20), $\overline{Acc}$(21-30), $\overline{Acc}$(31-40), and $\overline{Acc}$(41-50) our proposed method - II outperforms other methods with IMA of 0.963, 0.953, 0.937, 0.917, and 0.890 on CIFAR-10 dataset respectively and with IMA of 0.818, 0.794, 0.764, 0.730, and 0.694 on CIFAR-100 dataset respectively. At last, for degradation interval, $\overline{Acc}$(All), ”Ours - II” performs the best with an IMA of 0.933 on the CIFAR-10 dataset, and ”Ours - II” also outperforms other methods significantly with an IMA of 0.761 on CIFAR-100 dataset.

Additionally, we have shown a graphical representation of accuracy at all degradation levels for the CIFAR-100 dataset in Fig. 4 for the degradation of JPEG compression, salt-and-pepper noise (SAPN), Gaussian blur, and additive white Gaussian noise (AWGN) in (a), (b), (c), and (d), respectively.

Discussion: We analyze the performance of our proposed network on different datasets, degradation methods, degradation intervals, and different methods for image classification of degraded images. The benefit of the FA method [20] is that performance on clean images or lower degradation levels is highest, similar to the ”Clean” network as shown in Fig. 4. However, $\overline{Acc}$(All) of FA method [20] is often comparable to ”DEG” and ”DIST” methods as shown in Tables 4, 5, 6 and 7. Overall, our proposed method - II can distinctly outperform other methods for degradation types, i.e., Salt-and-pepper noise and additive white Gaussian noise due to the effective geometric-and-color and cut-and-delete DA, which increases the robustness and performance for degraded image classification without the need of additional feature extractors or estimators that have been used in previous works. Similarly, our proposed method II can outperform other existing methods on JPEG and Gaussian blur for most of the degradation intervals except the ”Clean Image” and a few higher degradation intervals.

4.6.2 Comparison on Tiny ImageNet dataset

To show the impact of our network on a larger dataset, we extended our experiments to the Tiny ImageNet dataset. We compared a total of five methods, i.e., ”Clean,” ”DEG,” ”DIST,” and our proposed methods ”Ours - I” [9], and ”Ours - II” as shown in Tables 8 and 9. We present the results of the Tiny ImageNet dataset on all four degradation methods, i.e., JPEG compression, salt-and-pepper noise, Gaussian blur, and additive white Gaussian noise, as discussed in previous sections. The Tiny ImageNet dataset experiments were all conducted on ResNet56 backbones rather



(a) JPEG compression

(b) Salt-and-pepper noise (SAPN)

(c) Gaussian blur

(d) Additive white Gaussian noise (AWGN)

Fig. 4: The accuracy of feature extractor based on ShakePyramidNet backbone for the degradation of JPEG, salt-and-pepper noise, Gaussian Blur, and AWGN in (a), (b), (c), and (d) respectively on CIFAR-100 dataset.

than ShakePyramidNet backbones for the following reasons: (1) To show the significance of our proposed networks on lightweight applications and (2) Due to the limited capacity of our GPU training environment. Also, the ResNet56 backbone is much smaller, i.e., 0.9M training parameters, compared to the ShakePyramidNet110 backbone's size of 28.5M training parameters.

Discussion: Our proposed method - II outperforms all other approaches at all degradation intervals except the $\overline{Acc}$(4.1-5) of Gaussian blur, as shown in Tables 8 and 9. Our proposed methods can substantially outperform the ”Clean” method performance on the ”Clean Image” interval for JPEG, SAPN, and AWGN degradation types. In fact, our proposed methods can substantially outperform on the lowest or highest degradation intervals where our proposed methods, in a few cases of CIFAR-10 / CIFAR-100 datasets, could not outperform other approaches such as ”Clean” or ”DEG.” That shows our proposed methods are more scalable on larger datasets like Tiny Imagenet on all degradation intervals and types. Besides, our proposed method ”Ours - II” is

Table 8: Interval mean accuracy for the feature extractor based on ResNet56 backbones with several degradation methods such as JPEG compression and salt-and-pepper noise on Tiny ImageNet dataset for different degradation intervals.

(a) JPEG

Degradation Interval	Clean	DEG	DIST	Ours - I	Ours - II
Clean Image	0.573	0.565	0.595	0.605	**0.612**
$\overline{Acc}$(1-20)	0.570	0.565	0.596	0.604	**0.612**
$\overline{Acc}$(21-40)	0.554	0.564	0.593	0.602	**0.610**
$\overline{Acc}$(41-60)	0.468	0.550	0.573	0.583	**0.588**
$\overline{Acc}$(61-80)	0.423	0.537	0.559	0.568	**0.576**
$\overline{Acc}$(81-100)	0.156	0.427	0.445	0.446	**0.457**
$\overline{Acc}$(All)	0.436	0.529	0.554	0.561	**0.569**

(b) Salt-and-pepper noise (SAPN)

Degradation Interval	Clean	DEG	DIST	Ours - I	Ours - II
Clean Image	0.573	0.570	0.580	0.601	**0.612**
$\overline{Acc}$(0.01-05)	0.369	0.570	0.580	0.601	**0.613**
$\overline{Acc}$(0.06-0.1)	0.153	0.571	0.579	0.601	**0.613**
$\overline{Acc}$(0.11-0.15)	0.071	0.568	0.579	0.600	**0.612**
$\overline{Acc}$(0.16-0.2)	0.036	0.567	0.578	0.599	**0.610**
$\overline{Acc}$(0.21-0.25)	0.021	0.565	0.577	0.599	**0.609**
$\overline{Acc}$(All)	0.147	0.568	0.579	0.600	**0.611**

Table 9: Interval mean accuracy for the feature extractor based on ResNet56 backbones with several degradation methods such as Gaussian blur, and AWGN on Tiny ImageNet dataset for different degradation intervals.

(a) Gaussian blur

Degradation Interval	Clean	DEG	DIST	Ours - I	Ours - II
Clean Image	**0.573**	0.472	0.567	0.571	0.572
$\overline{Acc}$(0.1-1)	0.415	0.472	0.567	0.568	**0.570**
$\overline{Acc}$(1.1-2)	0.067	0.469	0.549	0.547	**0.553**
$\overline{Acc}$(2.1-3)	0.025	0.456	0.513	0.512	**0.515**
$\overline{Acc}$(3.1-4)	0.017	0.428	0.469	0.471	**0.471**
$\overline{Acc}$(4.1-5)	0.016	0.395	**0.428**	0.424	0.427
$\overline{Acc}$(All)	0.117	0.445	0.507	0.506	**0.508**

(b) Additive white Gaussian noise (AWGN)

Degradation Interval	Clean	DEG	DIST	Ours - I	Ours - II
Clean Image	0.573	0.571	0.598	0.614	**0.621**
$\overline{Acc}$(1-10)	0.386	0.571	0.598	0.611	**0.617**
$\overline{Acc}$(11-20)	0.185	0.565	0.589	0.601	**0.607**
$\overline{Acc}$(21-30)	0.059	0.548	0.572	0.582	**0.588**
$\overline{Acc}$(31-40)	0.020	0.520	0.547	0.558	**0.566**
$\overline{Acc}$(41-50)	0.011	0.467	0.496	0.534	**0.537**
$\overline{Acc}$(All)	0.141	0.535	0.561	0.578	**0.584**

better than the proposed method "Ours - I" [9] method on the overall degradation interval $\overline{Acc}$(All) performance by 0.8%, 1.1%, 0.2%, and 0.6% on the degradation of JPEG compression, salt-and-pepper noise, Gaussian blur, and AWGN respectively. This fact shows that the robust data augmentation methods and cut-and-delete DA positioning help our proposed method - II to outperform our previous proposed method - I.

In summary, across all datasets in this study, the Ours-II proposed method is more effective by a reasonable margin than Ours-I for AWGN and SAPN across different degradation intervals. For JPEG and Gaussian blur, however, Ours-II performance is comparable. Therefore, we conclude that the Ours-II is slightly better than Ours-I.

5. Limitations

This study focuses on addressing specific known degradations and evaluating how robust data augmentation and knowledge distillation impact the training of classifiers for these degradations. However, in real-world scenarios, the types of degradations applied to an image are often unknown. While our study does not directly address unknown degradations, it provides valuable insights into the importance of knowledge distillation and data augmentation for classifying degraded images. These insights could serve as a foundation for achieving better generalization across various types of unknown degradations in future research.

6. Conclusion

This work establishes that our proposed method based on robust data augmentation methods such as geometric-and-color and cut-and-delete DA with knowledge distillation outperforms existing approaches on the image classification of degraded images with degradation such as JPEG compression, salt-and-pepper noise, Gaussian blur, and additive white Gaussian noise. We demonstrated that we could achieve better robustness and performance based on RandAugment for geometric-and-color and Random Erasing for cut-and-delete DA instead of introducing additional feature extractors or estimators modules, which typically leads to increased network size. We empirically demonstrate the efficacy of our proposed methods on the CIFAR-10 and CIFAR-100 datasets. Moreover, our methods exhibit significant superiority over existing approaches when applied to the Tiny Imagenet dataset, underscoring their effectiveness on realistic, larger datasets.

There are a few possible directions for future work in this area, such as (1) Single network that can handle several types of degradation [34] for unpaired clean/degraded images using domain adaptation to perform different computer vision tasks. (2) Design and study the impact of degraded images on the neural networks for object detection and semantic segmentation tasks. (3) Study effectiveness of specific data augmentations on performance improvement of individual degradations.

References

[1] N. Buch, S.A. Velastin, and J. Orwell, "A review of computer vision techniques for the analysis of urban traffic," IEEE Transactions on Intelligent Transportation Systems, vol.12, no.3, pp.920–939, 2011.

[2] J. Janai, F. Güney, A. Behl, and A. Geiger, "Computer vision for autonomous vehicles: Problems, datasets and state-of-the-art," Found. Trends Comput. Graph. Vis., vol.12, pp.1–308, 2017.

[3] A. Esteva, K. Chou, S. Yeung, N.V. Naik, A. Madani, A. Mottaghi, Y. Liu, E.J. Topol, J. Dean, and R. Socher, "Deep learning-enabled medical computer vision," NPJ Digital Medicine, vol.4, 2021.

[4] H. Tian, T. Wang, Y. Liu, X. Qiao, and Y. Li, "Computer vision technology in agricultural automation —a review," Information Processing in Agriculture, vol.7, pp.1–19, 2020.

[5] D.I. Patrício and R. Rieder, "Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review," Comput. Electron. Agric., vol.153, pp.69–81, 2018.

[6] Y. Wei, S.N. Tran, S. Xu, B.H. Kang, and M. Springer, "Deep learning for retail product recognition: Challenges and techniques," Computational Intelligence and Neuroscience, vol.2020, 2020.

[7] J. Yang, S. Li, Z. Wang, H. Dong, J. Wang, and S. Tang, "Using deep learning to detect defects in manufacturing: A comprehensive survey and current challenges," Materials, vol.13, no.24, 2020.

[8] S.P. Mohanty, J. Czakon, K.A. Kaczmarek, A. Pyskir, P. Tarasiewicz, S. Kunwar, J. Rohrbach, D. Luo, M. Prasad, S. Fleer, J.P. Göpfert, A. Tandon, G. Mollard, N. Rayaprolu, M. Salathe, and M. Schilling, "Deep learning for understanding satellite imagery: An experimental survey," Frontiers in Artificial Intelligence, vol.3, 2020.

[9] D. Daultani, M. Tanaka, M. Okutomi, and K. Endo, "Iliac: Efficient classification of degraded images using knowledge distillation with cutout data augmentation," Electronic Imaging, 2023.

[10] T. Devries and G.W. Taylor, "Improved regularization of convolutional neural networks with cutout," ArXiv, vol.abs/1708.04552, 2017.

[11] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," Proceedings of the AAAI Conference on Artificial Intelligence, vol.34, no.07, pp.13001–13008, Apr. 2020.

[12] E.D. Cubuk, B. Zoph, J. Shlens, and Q. Le, "Randaugment: Practical automated data augmentation with a reduced search space," Advances in Neural Information Processing Systems, pp.18613–18624, 2020.

[13] D. Cai, K. Chen, Y. Qian, and J.K. Kämäräinen, "Convolutional low-resolution fine-grained classification," Pattern Recognition Letters, vol.119, pp.166–171, 2019. Deep Learning for Pattern Recognition.

[14] T. Son, J. Kang, N. Kim, S. Cho, and S. Kwak, "Urie: Universal image enhancement for visual recognition in the wild," ECCV, 2020.

[15] Y. Pei, Y. Huang, Q. Zou, X. Zhang, and S. Wang, "Effects of image degradation and degradation removal to cnn-based image classification," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.43, pp.1239–1253, 2019.

[16] Y. Wang, Y. Cao, Z.J. Zha, J. Zhang, and Z. Xiong, "Deep degradation prior for low-quality image classification," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.11046–11055, 2020.

[17] Y. Pei, Y. Huang, and X. Zhang, "Consistency guided network for degraded image classification," IEEE Transactions on Circuits and Systems for Video Technology, vol.31, no.6, pp.2231–2246, 2021.

[18] Z. Yang, W. Dong, X. Li, M. Huang, Y. Sun, and G. Shi, "Vector quantization with self-attention for quality-independent representation learning," 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Los Alamitos, CA, USA, pp.24438–24448, IEEE Computer Society, jun 2023.

[19] K. Endo, M. Tanaka, and M. Okutomi, "Cnn-based classification of degraded images with awareness of degradation levels," IEEE Transactions on Circuits and Systems for Video Technology, vol.31, no.10, pp.4046–4057, 2021.

[20] K. Endo, M. Tanaka, and M. Okutomi, "Cnn-based classification of degraded images without sacrificing clean images," IEEE Access, vol.9, pp.116094–116104, 2021.

[21] S. Yang, W.T. Xiao, M. Zhang, S. Guo, J. Zhao, and S. Furao, "Image data augmentation for deep learning: A survey," ArXiv, vol.abs/2204.08610, 2022.

[22] E.D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q.V. Le, "Autoaugment: Learning augmentation policies from data," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.113–123, 2019.

[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," International Journal of Computer Vision (IJCV), vol.115, no.3, pp.211–252, 2015.

[24] H. Naveed, "Survey: Image mixing and deleting for data augmentation," ArXiv, vol.abs/2106.07085, 2021.

[25] G.E. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," ArXiv, vol.abs/1503.02531, 2015.

[26] H. Wang, S. Lohit, M.J. Jones, and Y. Fu, "What makes a "good" data augmentation in knowledge distillation - a statistical perspective," Advances in Neural Information Processing Systems, ed. A.H. Oh, A. Agarwal, D. Belgrave, and K. Cho, 2022.

[27] S. Yun, D. Han, S.J. Oh, S. Chun, J. Choe, and Y.J. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp.6022–6031, 2019.

[28] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," Advances in Neural Information Processing Systems 32, pp.8024–8035, Curran Associates, Inc., 2019.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.770–778, 2015.

[30] D. Han, J. Kim, and J. Kim, "Deep pyramidal residual networks," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.6307–6315, 2016.

[31] Y. Yamada, M. Iwamura, T. Akiba, and K. Kise, "Shakedrop regularization for deep residual learning," IEEE Access, vol.7, pp.186126–186136, 2018.

[32] K. Endo, M. Tanaka, and M. Okutomi, "Cnn-based classification of degraded images," Electronic Imaging, vol.2020, no.10, pp.28–1–28–7, 2020.

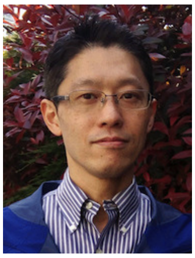
[33] K. Endo, M. Tanaka, and M. Okutomi, "Classifying degraded images over various levels of degradation," 2020 IEEE International Conference on Image Processing (ICIP), pp.1691–1695, 2020.

[34] D. Daultani and H. Larochelle, "Consolidating separate degradations model via weights fusion and distillation," Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops, pp.440–449, January 2024.

Dinesh Daultani received his B.Eng. degree in computer science from RGPV, Bhopal, India, in 2010, and his Master of Science in Information Systems from Illinois State University, the USA, in 2014. He is currently a Ph.D. student since 2021 in the Department of Systems and Control Engineering, Tokyo Institute of Technology. Additionally, he worked as a Research Scientist at Rakuten from 2018 to 2021 and as a Data Scientist at Woven by Toyota from 2021 to 2022.

Masayuki Tanaka received bachelor's and master's degrees in control engineering and a Ph.D. from the Tokyo Institute of Technology in 1998, 2000, and 2003, respectively. He was a software engineer at Agilent Technologies from 2003 to 2004. He was a Research Scientist at the Tokyo Institute of Technology from 2004 to 2008. He was an Associate Professor at the Graduate School of Science and Engineering, Tokyo Institute of Technology, from 2008 to 2023. He was a Visiting Scholar with the Department of Psychology, Stanford University, CA, USA, from 2013 to 2014. Since 2023, he has been a Professor at Graduate Major in Engineering Sciences and Design, Department of Systems and Control Engineering, School of Engineering, Tokyo Institute of Technology.

Masatoshi Okutomi received a B.Eng. degree from The University of Tokyo in 1981 and an M.Eng. degree from the Tokyo Institute of Technology in 1983. He joined Canon Research Center in 1983. From 1987 to 1990, he was a Visiting Research Scientist with the School of Computer Science at Carnegie Mellon University. He received his Ph.D. by dissertation from the Tokyo Institute of Technology in 1993. Since 1994, he has been with Tokyo Institute of Technology, where he is currently a Professor with the Department of Systems and Control Engineering, School of Engineering.



Kazuki Endo received a bachelor's degree in mathematics, a master's degree in industrial engineering and management, and a D.Eng. degree in systems and control engineering from the Tokyo Institute of Technology, Tokyo, Japan, in 1997, 1999, and 2022, respectively. He joined the Industrial Bank of Japan, Ltd., Tokyo, in 1999. Since 2022, he has been an Associate Professor with the Department of Business at Teikyo Heisei University, Tokyo, Japan.