Pattern Recognition 109 (2021) 107594

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

GridMix: Strong regularization through local context mapping

Kyungjune Baek[a,1], Duhyeon Bang[b,1,2], Hyunjung Shim[a,*]

[a] School of Integrated Technology, Yonsei University, 85, Songdogwakak-ro, Yeonsu-gu, Incheon, Republic of Korea
[b] SK T Tower, 65 Eulji-ro, Jung-gu, Seoul, Republic of Korea

ARTICLE INFO

Article history:
Received 10 May 2020
Revised 11 August 2020
Accepted 11 August 2020
Available online 14 August 2020

Keywords:
Deep learning
Network regularization
Data augmentation

ABSTRACT

Recently developed regularization techniques improve the networks generalization by only considering the global context. Therefore, the network tends to focus on a few most discriminative subregions of an image for prediction accuracy, leading the network being sensitive to unseen or noisy data. To address this disadvantage, we introduce the concept of local context mapping by predicting patch-level labels and combine it with a method of local data augmentation by grid-based mixing, called GridMix. Through our analysis of intermediate representations, we show that our GridMix can effectively regularize the network model. Finally, our evaluation results indicate that GridMix outperforms state-of-the-art techniques in classification and adversarial robustness, and it achieves a comparable performance in weakly supervised object localization.

1. Introduction

With the emergence of large-scale datasets and advanced model architectures, deep neural networks have delivered remarkable performances in various computer-vision tasks such as classification [1,2], localization [3,4], and segmentation [5,6]. To achieve a high prediction accuracy, modern neural network models use considerably more model parameters than training examples. This overparameterization leads the model to memorize all the training data, so that it overfits them. This overfitting results in (1) a severe performance degradation in the predication of unseen data and (2) failure to handle samples that deviate slightly from the data distribution, such as adversarial examples [7,8]. To address these challenges, various model-regularization techniques have been actively studied, the most representative of which include dropout [9] and data augmentation.

As a training strategy for model regularization, dropout deactivates several randomly selected parameters during training and uses all the parameters during inference. The effect of dropout can be interpreted as an ensemble of subnetworks, and it substantially improves the generalization of models. However, the performance gain of dropout is limited to fully connected networks, which are not effective for many computer-vision tasks that utilize convolutional neural networks (CNN) [10]. Recently, regional dropout techniques have overcome these limitations by erasing several regions in images or feature maps, e.g., the most discriminative parts or random regions, instead of model parameters. Regional dropout helps the model consider the overall regions of the input or feature map for making the decision, thereby improving the generalization of CNN models [11,12].

Recently, widespread studies have been conducted on various regularization techniques under the principle of data augmentation. Among these techniques, CutOut [12] and Hide-and-Seek (HaS) [11] utilize regional dropout for data augmentation and deliver an impressive performance gain in the weakly supervised object localization task. These techniques erase subregions of the input images and use them as augmented data. They then map these augmented data to the same output as the input. Consequently, the network model effectively improves generalization because the prediction of the output by focusing only on the subregions of the input is prohibited. Including CutOut and HaS, standard data augmentation methods generate additional data for the same class. A recent technique called MixUp [13] suggests also generating training examples in-between different classes having soft labels. Because MixUp linearly combines two images for generating both the data and the labels, it encourages the model to learn the rich visual features. Thus, it has a clear advantage over standard data augmentation methods, which force the model to discard the visual features for generating the augmented data. Because the modern network models inevitably suffer from a lack of data, in general, the idea of mixed data can be a useful trick for augmenting the data.

[*] Corresponding author.
E-mail addresses: bkjbkj12@yonsei.ac.kr (K. Baek), duhyeonbang@sktbrain.com (D. Bang), kateshim@yonsei.ac.kr (H. Shim).
[1] Equal contribution.
[2] This work was done during his doctoral course.

https://doi.org/10.1016/j.patcog.2020.107594
0031-3203/© 2020 Elsevier Ltd. All rights reserved.

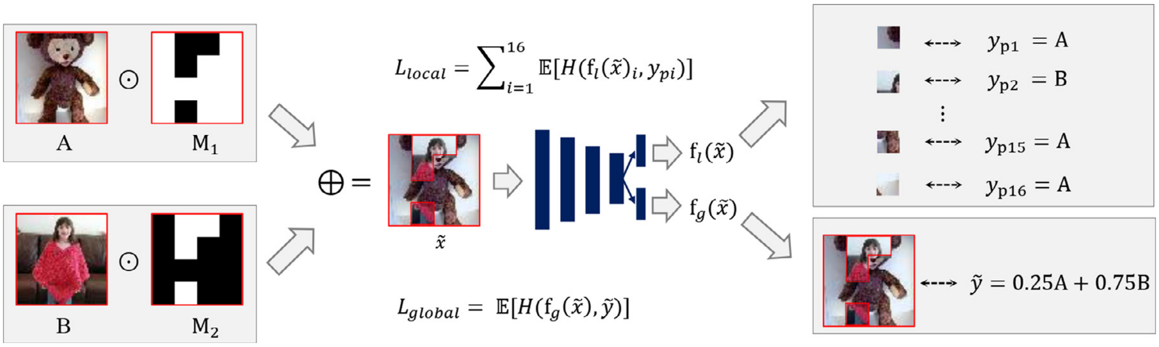

A
⊙
M_1
B
⊙
M_2
$\oplus =$
$L_{local} = \sum_{i=1}^{16} \mathbb{E}[H(f_l(\tilde{x})_i, y_{pi})]$
$\tilde{x}$
$f_l(\tilde{x})$
$f_g(\tilde{x})$
$L_{global} = \mathbb{E}[H(f_g(\tilde{x}), \tilde{y})]$
$y_{p1} = A$
$y_{p2} = B$
⋮
$y_{p15} = A$
$y_{p16} = A$
$\tilde{y} = 0.25A + 0.75B$

Fig. 1. Training procedure of GridMix. $\tilde{x}$ represents a mixed image generated from A and B, $\tilde{y}$ represents a mixed label, y_{pi} represents a patch label at the i_{th} cell, and $f(\cdot)$ indicates our network model. We define the global context loss to model f_g and the local context loss to model f_l, where the two functions share all the layers before the last, classification, layer. $f_l(\tilde{x})_i$ outputs the predicted class label of the i_{th} cell. Each predicted class label is used to calculate the cross-entropy loss with y_{pi}.

In the same direction, CutMix [14] utilizes the advantages of regional dropout and MixUp for improving the generalization of the classifier. CutMix also discards the patch from the input image to prevent overfitting to particular local visual features, as does Cutout. However, because the model is data-hungry, CutMix replaces the dropped patch with a patch from another image, as in MixUp, which produces the augmented data by mixing two different images. Moreover, CutMix assigns the ground-truth label for a mixed image by computing the mixture of the two labels, where the ratio of the two images is the weight for their two labels. As a result, it utilizes the effect of the regional dropout of CutOut and the data-label augmentation of MixUp. By using the advantages of both approaches, CutMix achieves a significant gain in generalization.

Although their performance is impressive, existing regularization techniques consider only the global context, such as an image label, as the mapping criteria. Utilizing this global context constraint, the network tends to focus solely on a few subregions of an image for improving the prediction accuracy. Specifically, a classifier trained with image-level labels naturally focuses on a single or a few of the most discriminative parts rather than the integral object. The standard regularization techniques, such as CutOut or CutMix, aim to mitigate this issue by dropping or replacing subregions; however, as a result, the attention regions of feature maps can be erroneous (i.e., although these regularization methods can expand the attention of feature maps, the change in the feature maps is not necessarily accurate). Instead, undesirable regions containing objects that are not of interest can be captured. Through our experiments, we show that, whereas CutMix induces these inaccurate attention regions, GridMix leads to accurate attentions in feature maps.

In this paper, we introduce the concept of local context mapping and combine this idea with a data-augmentation method that utilizes grid-based mixing, called GridMix. Specifically, we divide the input image into $N \times N$ grid cells and, at every cell of the grid, a patch is randomly selected from one of two images to fill that cell. This grid-based method of data augmentation allows the mixture of the two images to be discontinuous; see Fig. 1. In contrast, in CutMix, only one continuous patch is from a different image.

Then, we define two constraints: local context and global context mappings. Using local context mapping, the model predicts to which of the two classes each patch belongs. Using global context mapping, the model estimates the mixed label from the mixed image, which is encoded by the overall ratio of the two images. The key idea of GridMix is to prevent the model from (global) overfitting by inducing overfitting at every local patch. This interesting idea can be simply realized by predicting the label per patch. In addition, the method of predicting the class label per patch is akin to the self-supervised learning strategy in that it transforms the data in a predefined manner, and then, its transformation parameters (i.e., a class label per patch) are utilized as artificial labels. Although the idea of local context mapping is extremely simple, it forces the model to examine entire regions of the image. Thus, our strategy effectively expands the attention region of the model. Moreover, whereas regularization techniques that utilize only the global context cannot prevent erroneous changes in the attention regions of feature maps, GridMix utilizes the local context for penalizing inaccurate attentions. As a result of all these useful properties, GridMix leads to an improved generalization by considering the accurate integral object for prediction.

As a strong regularization technique, GridMix significantly improves the classification and localization accuracies of the baseline network. More importantly, it achieved a new state-of-the-art performance in the classification task on the CIFAR100 [15] and Tiny-ImageNet [16] datasets. Moreover, our results indicate that GridMix is more robust than other regularization techniques against natural adversarial examples.

Our main contributions are summarized below.

- We propose a novel regularization method that consists of two local constraints: local data augmentation by grid-based image mixing and local patch mapping constrained by patch-level labels.
- We conducted various case studies, including comparisons with the state-of-the-art regularization techniques on three different tasks (i.e., classification, weakly-supervised localization, and robustness) using various benchmark datasets (CIFAR-100, Tiny-Imagenet, CUB-200-2011, TinyImagenet-A, and ILSVRC2012) and with various models (VGG, ResNet, pre-activate ResNet, and wide ResNet, ResNext).
- Through our analysis of intermediate representations, we verify that the two local constraints can effectively regularize the network model.

2. Related work

2.1. Dropout and beyond

Dropout [9] is an effective regularization technique that deactivates several nodes of fully connected networks during training but uses all the nodes for inference. Thus, it effectively prevents neural networks from overfitting on the training data.

Because the original version of dropout is not suitable for CNNs, several variants of dropout intended to handle CNNs were proposed. We categorize them as regional dropout techniques. Spatial dropout [10] first indicated the limitation of dropout—the original dropout does not effectively deactivate the node in a CNN because of the strong spatial correlation of network activation. Then, spatial dropout was proposed to successfully deactivate the activations of the fully convolutional network. HaS [11] divides the image into a grid and randomly drops patches. It then fills the dropped patches by the mean value of a mini-batch. Using this simple training tactic, HaS improves the performance of weakly supervised object localization. CutOut [12] improves the classification performance of the network by blocking the activation propagation from the input image to the feature map. This is achieved by randomly zeroing out a rectangle at the input image level during training.

Regional dropout typically improves the generalization performance by penalizing the network focusing only on the small subregion (the most discriminative part). As a byproduct, the network earns the performance benefit because it implicitly learns the relationship between the remaining parts [17]. These methods can be considered data-augmentation methods, where the augmented data constitute a dropped version of the training data. GridMix and HaS are similar in that they both utilize an $N \times N$ grid and randomly drop its cells. However, GridMix suggests filling the dropped region with a patch taken from other training data. This encourages the model to learn the rich visual features.

2.2. Auxiliary supervision for improving the model performance

Recently, auxiliary supervision methods, such as transformations applied to the input data, have been actively utilized in various applications. RotNet [18] assigns different rotations to the input image and uses the rotation information as self-supervision to learn the unsupervised classifier. In self-supervised generative adversarial networks (GANs) [19], the idea of RotNet to train the discriminator was adopted. Specifically, the discriminator is forced to predict the rotation angle applied to the image during adversarial training. As a result, they significantly improve the quality of image generation (production of realistic shapes, in particular).

As a new data-augmentation strategy, mixing the training data is also considered a model regularization technique. This new strategy is different from conventional data-augmentation methods in that it assigns new labels for mixed data. (The conventional data-augmentation method assigns the same label for augmented data.) MixUp [13], CutMix [14], and manifold MixUp [20] are the most representative techniques of this type. MixUp trains the network with mixed images and mixed labels, where the mixed images are generated by the convex combination of two images at the pixel level. The mixed labels are calculated by combining the two labels with the ratio of the two images in the mixed images. MixUp improves the performance of the network by enforcing the linear behavior of the network in-between training samples. Manifold MixUp applies MixUp to the intermediate layer and thus can be considered an extension of MixUp. CutMix is aimed to utilize the advantages of both CutOut and MixUp—it first conducts the regional dropout as CutOut does, but it fills the dropped region with a region from a second image.

The proposed GridMix is similar to MixUp in that mixed images and their corresponding labels are also generated, and they are used as auxiliary supervision. Moreover, similar to CutMix, our method erases the local regions, as does regional dropout, and replaces them with regions from another image. However, unlike any previous method, GridMix introduces local context mapping and uses it as additional natural supervision. Thus, the network is guided to consider both the global (i.e., image-level labels) and the local contexts (i.e., patch-level labels). In our experimental studies, we observed that GridMix produces accurate attention regions and improves the generalization in various tasks.

3. GridMix toward local context mapping

3.1. Motivation

Recently, network regularization techniques that use the concept of data augmentation have shown promising results in terms of improving generalization in various decision tasks. Among them, we focus on MixUp [13] and CutMix [14], which, despite being simple data-augmentation techniques, deliver an impressive performance. Motivated by their success, we propose GridMix, which performs dropout and data augmentation simultaneously to improve the CNN classifiers performance. GridMix is similar to MixUp and CutMix in that it produces data-label pairs by mixing two images and their labels using the same ratio.

However, we highlight that all the existing techniques, including MixUp and CutMix, only consider a global context constraint as supervision, such as an image-level label. To maximize the prediction accuracy, the model naturally selects as its focus a few subregions that are most discriminative to reach a correct decision. Existing regularization techniques may prevent a model from focusing only on a single subregion (e.g., the most discriminative part) by randomly dropping subregions during training [13,14]. Although they successfully expand the attention regions of feature maps, these schemes do not guarantee that the expanded attention maps accurately cover the target object. In fact, we have frequently observed that their attention maps are expanded by covering an undesirable object or background.

To address the limitations mentioned earlier, we propose GridMix, which utilizes the local context as additional supervision. GridMix combines two images at the patch level, and every patch in the mixed image is randomly selected from one of the two images. Utilizing this grid-based mixing, we define an additional constraint that identifies the class label at every local patch, namely a local context constraint.

For this purpose, we divide the image into an $N \times N$ grid. Then, a new image is generated by selecting a local patch of one of the two images stochastically with a prefixed probability rate. The label for this generated image is computed by mixing the two original labels in proportion to the number of cells, as in MixUp and CutMix.

3.1.1. GridMix is a stronger data-augmentation method

Our grid-based mix method can be interpreted as a stronger data-augmentation technique than CutMix because it can drop multiple discontinuous parts, thereby utilizing more diverse and challenging augmented data. The results of our ablation study confirm that the adoption of grid-based dropout alone as regional dropout can be an effective regularization technique for the network model (e.g., it improves the classification accuracy for unseen data).

3.1.2. GridMix utilizes patch labels as additional self-supervision

We take advantage of a grid-based mix as a clue for utilizing local context mapping. Because each cell in the grid is from one of the two images, a new problem of identifying the class label of each cell can be defined. Specifically, we force the network to predict not only the mixed label of the mixed input but also the label at each cell. This local context mapping can be used as an additional constraint to prevent a situation where the network overly depends on the most discriminative part of the entire image for classifying the image label. Because a network trained with local context mapping must predict the class label per cell, the model naturally extracts at least one of the most discriminative parts from each cell. Consequently, the model learns to extract the most discriminative features from all the cells, which are evenly distributed over the entire image. This is equivalent to extracting multiples of the most discriminative features covering the entire input image. The role of this local constraint can be considered as that of a regularizer that forces the activation for the input to spread over the entire image. In another perspective, the idea of local constraint can be interpreted as an attempt to utilizing multiple features to represent the image [21].

3.1.3. Why is GridMix better than CutMix or CutOut?

We claim that CutMix and MixUp cannot prevent the network from establishing an incorrect mapping from the image patch to the label. As compared to regional dropout-based regularization techniques, such as CutOut and CutMix, GridMix can not only expand the attention regions of feature maps but also guide their attentions more accurately by predicting patch-level labels (i.e., the local context mapping penalizes the model for paying attention to an undesirable object or to background regions). For example, suppose that we have a mixed image generated by replacing the head of a cat with that of a dog with the regional ratio of [0.5, 0.5]. CutMix cannot penalize the case where the network predicts [0.5, 0.5] by interpreting *the legs of a cat* as *the legs of a dog*. In contrast, our local constraint enforces correct mapping between the part of the object and the label.

The idea of the local constraint can also be interpreted as alleviating overfitting at the image level by inducing overfitting at the patch level (all the cells). Subsequently, we describe the training strategy for GridMix.

3.2. Algorithm

GridMix is applied only during training and is deactivated during the test phase. Thus, the testing phase is identical to that of a vanilla model. During training, the procedure of GridMix is similar to that of CutMix in that it generates a new image by replacing the dropped region with a subregion of a second image and determines the label for the new image by the ratio of the two images. Unlike existing techniques, GridMix utilizes a local context constraint by introducing grid-based regional dropout and local context mapping loss. Specifically, GridMix leads the network to predict the global context (i.e., the image label of the mixed image) and the local context (i.e., the class label of each cell) simultaneously. We illustrate the training procedure of GridMix in Fig. 1.

Formally, $x \in \mathbb{R}^{W\times H\times C}$ is a training image, y is an image label, N is the grid size, $\mathbf{M} \in \{0, 1\}^{N \times N}$ is the $N \times N$ binary mask indicating the class labels of all cells, and $\mathbf{M_x} \in \{0, 1\}^{W \times H}$ is the binary mask, the size of which is the same as that of the training image. We randomly select two training images $\{x_A, x_B\}$ and their labels $\{y_A, y_B\}$ and then compute new training data $\tilde{x}$ and label $\tilde{y}$ by combining $\{x_A, x_B\}$ and $\{y_A, y_B\}$. The procedure of GridMix can be written as follows.

$$
\begin{aligned}
&\mathbf{M}(i, j) \sim \mathrm{Ber}(p), \quad \lambda = \sum_{i,j} \mathbf{M}(i, j)/N^2 \\
&\tilde{x} = \mathbf{M_x} \odot x_A + (\mathbf{1} - \mathbf{M_x}) \odot x_B \\
&\tilde{y} = \lambda \odot y_A + (\mathbf{1} - \lambda) \odot y_B \\
&y_p = \mathbf{M} \odot y_A + (\mathbf{1} - \mathbf{M}) \odot y_B \qquad (1)
\end{aligned}
$$

where Ber is the Bernoulli distribution with parameter p, $\odot$ is element-wise multiplication, and y_p is an $N \times N$ label matrix, where each element of y_p is denoted by y_{pi} representing the class label of each cell. MixUp and CutMix respectively use the Beta distribution or uniform distribution to sample stochastically the ratio of the mixture (λ). Meanwhile, GridMix obtains a binary mask by stochastically sampling the value of each cell from the Bernoulli distribution with a fixed parameter $p = 0.8$. Then, two images in the same mini-batch are randomly sampled, and the mixed image ($\tilde{x}$), its corresponding image label ($\tilde{y}$), and its patch-level labels (y_p) are then generated, as represented in Eq. (1). Using $\{\tilde{x}, \tilde{y}, y_p\}$, the objectives of the network are defined as

$$\mathcal{L}_{total} = \mathcal{L}_{global} + \gamma \mathcal{L}_{local} \qquad (2)$$

$$
\begin{aligned}
\mathcal{L}_{global} &= \mathbb{E}[\mathcal{H}(f_g(\tilde{x}), \tilde{y})] \\
&= \lambda \cdot \mathbb{E}[\mathcal{H}(f_g(\tilde{x}), y_A)] + (1 - \lambda) \cdot \mathbb{E}[\mathcal{H}(f_g(\tilde{x}), y_B)] \qquad (3)
\end{aligned}
$$

$$\mathcal{L}_{local} = \mathbb{E}\left[\sum_i \mathcal{H}(f_l(\tilde{x})_i, y_{pi})\right] \qquad (4)$$

γ is empirically set to 0.15 and $\mathcal{H}$ represents the cross-entropy function, *i.e.* $\mathcal{H}(p, q) = -\int_x p(x) \log q(x) dx$. f_l and f_g are designed as the output of the same network at different layers. f_l is the $N \times N$ matrix, where each component of f_l indicates the predicted class label of the patch. f_g provides a predicted label for the input image. f_l and f_g share the same feature extractor before the classification layer. f_g is the output of the same route of the original network, and f_l is the output of the additional branch starting at the last layer of the feature extractor. $\mathcal{L}_{global}$ serves as a global constraint because it leads the network to predict the image label from the entire image. Specifically, we define $\mathcal{L}_{global}$ as the weighted sum of two cross-entropy losses. Meanwhile, $\mathcal{L}_{local}$ represents a local constraint because it induces the network to extract the patch-level label y_p for each patch. Using $\mathcal{L}_{local}$, we can enforce the network to focus on the most discriminative parts per patch, which leads to consider the less discriminative parts in the perspective of the entire image as well. By doing so, $\mathcal{L}_{local}$ helps subsiding the regional overfitting problem–focusing only on the most discriminative parts in prediction.

4. Experiments

In this section, we first show the advantage of GridMix on multiple tasks, such as classification and weakly supervised object localization, and its robustness to adversarial examples. Moreover, we analyze its intermediate representations to investigate its characteristics.

Evaluation protocols For the classification task, we chose VGG [22] with batch normalization (BN) [23] and ResNet-type networks (ResNet [24], PreResNet [25], Wide-ResNet [26], and ResNext [27]) as the backbone networks. They were selected to show that the proposed training strategy is useful for various backbone networks. The CIFAR100 [15], TinyImageNet [16], and ILSVRC2012 datasets [28] were used because they are the representative benchmarks for classification tasks. Specifically, we trained VGG19 with BN, ResNet, PreResNet, and WRN on the CIFAR100 dataset. For TinyImageNet, VGG11/19 with BN, ResNet, and ResNext were adopted. For the classification on the ILSVRC2012 dataset, we used ResNet50 as the backbone network. Note that the empirical tendency of TinyImageNet is consistent across different model architectures, and its data statistics must have strong correlations with that of ILSVRC2012. Thus, in our opinion, the results of ResNet50 can constitute a representative performance for the ILSVRC2012 dataset. For the weakly supervised object localization task, the CUB-200-2011 dataset [29] was used, and VGG16 with global average pooling was chosen as the backbone model. Finally, we evaluated the robustness of the trained model using adversarial examples. Specifically, instead of an adversarial example created through optimization, we utilized natural adversarial examples (ImageNet-A) collected by the authors of [30].

Protocol for analysis For the performance analysis, we utilized penultimate layer activations [31] for visualizing the properties of GridMix and other methods. The purpose was to show that GridMix effectively alleviates overfitting and helps in regularizing the

Table 1
Comparisons of the baseline model, MixUp, CutMix, and GridMix on the image classification task. The number following the network name indicates the number of layers (i.e., the depth). In the case of WRN, the two numbers represent the depth and the width, respectively. The notations "w. and "wo. are the abbreviations of with and without, respectively. The bold text indicates the best performance in comparison with the competitors; this indication is used throughout the paper.

	Network	Baseline	MixUp	CutMix	GridMix wo. Eq. (4)	GridMix w. Eq. (4)
CIFAR100	VGG19BN	73.95	75.37	74.20	76.05	**77.09**
	ResNet101	75.71	77.83	77.74	78.30	**78.48**
	PreResNet101	76.50	77.39	78.15	79.17	**79.43**
	WRN28-4	79.33	80.44	81.11	81.11	**81.49**
Tiny Imagenet	VGG11BN	50.41	52.88	53.36	53.85	**59.16**
	VGG19BN	60.13	62.66	61.89	62.22	**63.34**
	ResNet50	65.23	67.34	67.22	68.38	**68.88**
	ResNext50 (32x4d)	66.66	68.38	66.81	68.80	**69.12**

network. In particular, the similarity of the activation distributions of the training and validation samples was used to demonstrate whether the model suffers from overfitting. Moreover, the distance from other class distributions shows the effectiveness of the model in distinguishing each class (i.e., it shows its regularization and robustness). In addition, we investigated the spatial attention of the trained model to observe the effectiveness of the regularization technique in leading the model to cover the target object accurately and entirely. All the experiments for our analysis were conducted using VGG11 with the BN network on TinyImageNet.

4.1. Classification

4.1.1. Effects of local context mapping

GridMix consists of two submodules. The first can be considered data augmentation by means of grid-based mixing; it still utilizes the global context (i.e., an image-level label) for optimizing the model parameters, as shown in Eq. (3). The second module utilizes the local context loss (i.e., a patch-level label) defined by Eq. (4). To clarify the effect of each module, we compared GridMix with and without local context loss. The reader may be concerned about the scenario where the network is trained only with local context loss. The network trained with the local context loss alone, unfortunately, does not converge at all. This is because it can be damaged by the poor learning signals that occur when the object in an input image is small and many patches capture the background, representing noisy data. When applied with the global context loss in Eq. (3), the local context loss can behave appropriately as a regularizer; the global context loss resolves the convergence issue of the local context loss, and the local context loss alleviates overfitting of the global context loss.

4.1.2. Comparison of GridMix with the state-of-the-art methods

To evaluate GridMix as compared to existing methods, we chose MixUp [13] and CutMix, which are the state-of-the-art regularization techniques, as the comparison methods. Table 1 presents the results obtained by the various models across the datasets. In all the cases, GridMix with/without the local context loss outperforms existing methods. In particular, when we strengthen the constraint by adding the local context loss, the trained models achieve a higher accuracy level. This is because the local context loss induces more accurate and expanded attention for the target object. As a result, the network is further regularized by considering all the regions of the entire image with more accurate attentions. Further, we highlight that on CIFAR100, our hyperparameter set is searched only once on VGG19BN and the same set is repeatedly used for all the remaining datasets and model architectures. Unlike GridMix, other state-of-the-art techniques require an extensive hyperparameter search per model per dataset because their results are sensitive to the selection of hyperparameters. In our opinion, our achievement is impressive because GridMix can still achieve a new state-of-the-art performance even with a single set of hyperparameters.

Table 2
Evaluating the scalability on a large-scale dataset. The table compares classification performance of the baseline model, MixUp, CutMix, and GridMix on the ILSVRC2012 dataset. The bold text indicates the best performance in comparison with the competitors; this indication is used throughout the paper.

	Network	Baseline	MixUp	CutMix	GridMix
ILSVRC 2012	ResNet50	74.08	77.37	78.20	**78.25**

To show the scalability of GridMix on a large-scale dataset, we conducted a classification experiment using the ILSVRC2012 dataset. We used the same hyperparameters searched on CIFAR100 for the ILSVRC2012 experiment. Table 2 reports on the classification accuracy rates of GridMix and the state-of-the-art techniques on the ILSVRC2012 dataset. GridMix slightly outperforms CutMix, even without the hyperparameter search. Because the hyperparameter search on a large-scale dataset such as ILSVRC2012 requires exceptionally large computing resources, performance sensitivity to the selection of hyperparameters is, in particular, undesirable. Therefore, we intentionally compared GridMix under minimally selected hyperparameters with CutMix under extensively selected hyperparameters. Finally, our results indicate that GridMix outperforms CutMix.

4.1.3. Analysis of the attention maps of trained networks

To gain a better understanding of the regularization effects of GridMix, we computed the attention map of the input data and compared it with the maps obtained by MixUp and CutMix. Because the attention map is known to encode the attention of the model, in our opinion, the attention of a successfully regularized model must cover the entire target area correctly [32].

Fig. 2 presents the attention maps extracted from MixUp, CutMix, and GridMix. To compute the attention map, we followed [33]. The size of the attention regions obtained from the different regularization techniques is not widely different, but the locations of the regions do differ. Both Mixup and CutMix not only focus on the regions for the target object but also consider the background regions. In contrast, the attention map of GridMix accurately covers the target object, and therefore, its attention coverage for the target object is more accurate than that of the other maps and is the best of all the three methods. For example, in the third row of Fig. 2, the networks trained with MixUp and CutMix focus on the surroundings of the front leg of the camel, but the network trained with GridMix covers not only the front leg but also the backside and the hump of the camel. In other words, GridMix covers the integral of the target object more accurately than the other models. This consistently holds for other examples.

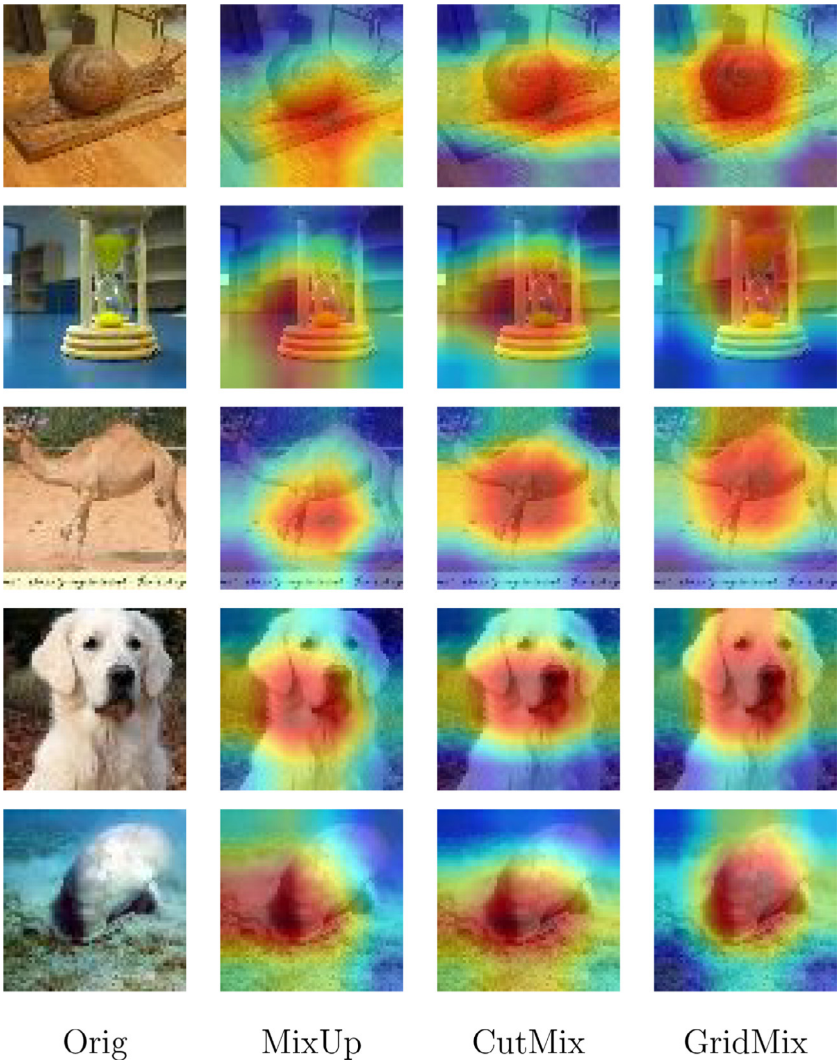


Fig. 2. Visualization of the attention maps produced by the networks trained with the standard data augmentation (Orig), MixUp, CutMix, and GridMix. GridMix consistently captures the target object more accurately than the other methods.

These results support our claim that GridMix leads to correct and expanded attentions in the feature maps. We believe that the local context serves as additional supervision and induces an accurate mapping even for the local patch. This observation leads us to conclude that GridMix constitutes a successful regularization strategy.

4.1.4. *Ablation study*

GridMix has two hyperparameters: the total number of grid cells (i.e., $N \times N$) and a parameter of the Bernoulli distribution. By performing evaluations under various hyperparameter settings, we could analyze the influence of each hyperparameter on the performance. Table 4 details the accuracy of VGG19BN [22] on the CIFAR100 [15] dataset. First, we conducted the ablation study on the different grid sizes and observed the similar performance with 2 × 2, 4 × 4, and 8 × 8. For the case of the large grid size (e.g., 16 × 16), the performance is significantly degraded. For 16 × 16, each cell contains only four pixels because the image size is 32 × 32. Therefore, we conjecture that each cell is too small to contain the discriminative features. As a result, the local context matching introduced by Eq. (4) no longer serves as a strong regularization term but rather degrades the overall training. When comparing the effects of 2 × 2, 4 × 4 and 8 × 8, although 4 × 4 is finally selected, the performance gaps are marginal. That is, GridMix is relatively robust to the choice of hyper-parameter. In practice, we recommend to set the size of grid cells as the size of feature map at the last convolutional layer.

We also studied the impact of the mixing probability. We changed the probability(p) within the range from 0.5 to 0.9. From 0.5 to 0.8, the accuracy increases as the probability increases, however, the accuracy drops at $p = 0.9$. When $p = 0.5$, the input images are most likely to be mixed. That means, the network is unlikely to observe the entire object at once thus less rely on the global structure of the object for the prediction. Consequently, the effect of Eq. (3) significantly decreases as p approaches to 0.5. Contrarily, when $p = 0.9$, the input images are less likely to be mixed so that the training images are similar to the original input images. As a result, the regularization effect of Eq. (4) is reduced and then the accuracy is subsequently lower than the case $p = 0.8$. (but still higher than the baseline) For a similar reason, the designers of MixUp [13] also chose Beta distribution with (0.4, 0.4) as their hyperparameters. Based on this ablation study, we set a 4 × 4 grid and $p = 0.8$ for the Bernoulli distribution in all our experiments.

4.1.5. *Time complexity*

Although we clearly showcase the advantage of GridMix in terms of improving the prediction accuracy, it is important to check whether or not GridMix requires excessive training over-

Table 3
Iterations per second and total training time for each model.

Model	MixUp	CutMix	GridMix
VGG19BN	9.5 it/s (8231 s)	9.0 it/s (8688 s)	8.5 it/s (9200 s)
ResNet101	4.0 it/s (19550 s)	3.8 it/s (20,578 s)	3.5 it/s (22,342 s)

Table 4
Ablation study for two hyperparameters on the image-classification task. We changed the size of the grid (i.e., $N \times N$) and the parameter of the Bernoulli distribution for generating the mixed image. In this experiment, VGG19 was utilized as the backbone network. Acc. indicates the classification accuracy. The bold text indicates the best performance in comparison with the competitors; this indication is used throughout the paper.

$N \times N$	2×2	4×4	8×8	16×16	Random	Baseline
Acc.	75.82	**76.05**	75.82	71.96	73.28	73.95
Prob	0.5	0.6	0.7	0.8	0.9	–
Acc.	71.81	73.18	75.93	**77.13**	75.78	–

heads. For analyzing the time complexity in training, we can estimate the big-O notation of the algorithm. In case of GridMix, generating a grid tensor corresponds to the largest growth rate. Because we utilize the Kronecker product to create a grid, we can derive the time complexity of GridMix by calculating that of the Kronecker product. We assume that the input image is $M \times M$ and the grid size is $N \times N$. Then, the time complexity of the Kronecker product is $O((N \cdot M/N) \cdot (N \cdot M/N)) = O(M^2)$, because the Kronecker product is based on the matrix multiplication. Considering the batch-based operation, the time complexity of GridMix for the given batch size is $O(\texttt{batch\_size} \cdot M^2)$. From this analysis, we conclude that 1) the time complexity of GridMix is proportional to the batch size and the square of the input size and 2) the training overheads of GridMix are far smaller than that of the forward and backward operation of a network.

Additionally, we measure the total training time of MixUp, CutMix and GridMix to quantify the training overheads. We train VGG19BN and ResNet101 on CIFAR100 using the above three methods, and then measure the average iterations per second. In this experiment, we set the batch size to 128– the total iterations per epoch are 391 for CIFAR100. Table 3 reports the average iterations per second and the total training time. Note that the larger *it*/*sec* and the smaller *sec* indicate the faster algorithm. GridMix takes about 11% (15 min in our hardware configuration) longer than MixUp, which is the fastest method among three methods on VGG19BN. We believe that such a overhead is acceptable in practice.

4.2. Weakly supervised localization

For weakly supervised object localization, we used a benchmark dataset, CUB-200-2011 [29], and trained the classifier only with image-level labels. For the quantitative evaluation, we used three standard metrics: GT-Known localization accuracy (*GT-Known Loc*), Top-1 Classification accuracy (*Top-1 Cls*), and Top-1 Localization accuracy (*Top-1 Loc*). *GT-Known Loc* measures the intersection over union (IoU) between the ground truth bounding box and its estimated bounding box. Then, the estimated bounding box is considered correct if the IoU is greater than or equal to 0.5. *Top-1 Cls* measures whether the ground-truth label of the image and the predicted label are the same. *Top-1 Loc* considers a result correct if both *GT-Known Loc* and *Top-1 Cls* are correct. The experimental results are presented in Table 5.

Table 5
Comparisons of the baseline model, MixUp, CutMix, and GridMix on the weakly supervised object localization task. “w. and ” wo. are the abbreviation of with and without, respectively. Note that the experiment was conducted using the VGG16 network trained on the CUB-200 dataset. The bold text indicates the best performance in comparison with the competitors; this indication is used throughout the paper.

Metric	Baseline	MixUp	CutMix	GridMix wo. Eq. (4)	GridMix w. Eq. (4)
Top-1 Cls	72.56	73.82	**74.80**	74.75	73.99
GT-Known Loc	60.08	54.77	69.83	69.10	**71.07**
Top-1 Loc	44.85	42.54	**55.58**	55.01	55.29

We observe that the performances of CutMix and GridMix are similar according to *Top-1 Loc*. Specifically, CutMix outperforms GridMix according to *Top-1 Cls*, but according to *GT-Known Loc*, GridMix achieves a better performance. This is because GridMix invokes the activation of all the cells, so that the heat map covers the entire object more accurately. This tendency is in agreement with the analysis of attention maps presented in Fig. 2.

Although it is advantageous to capture the integral extent of the object to improve the generalization in ordinary cases, this can considerably degrade the classification accuracy for fine-grained datasets. This is mainly due to the characteristics of a fine-grained dataset as considering the integral object is not helpful but rather confusing for label prediction. For example, on the CUB-200-2011 dataset, considering the entire body of a red-winged blackbird is not helpful for distinguishing it from a yellow-headed blackbird. This trade-off relationship between localization and classification is a well-known issue in object localization and has been discussed in previous studies [4,11]. This trade-off is of particular concern when a fine-grained dataset is being handled.

Because of this trade-off relationship, we observe that GridMix exhibits slightly lower *Top-1 Cls* and slightly higher *GT-Known Loc* values than CutMix. Overall, we observe that GridMix delivers a performance comparable to that of CutMix in the weakly supervised localization task.

4.3. Robustness

To evaluate the robustness of the network, we employed the ImageNet-A dataset introduced in Gilmer and Hendrycks [30]. ImageNet-A contains challenging samples (e.g., the failure cases of ResNet50) and thus can act as natural adversarial examples. We note that ImageNet-A and TinyImageNet [16] are different subsets of the ILSVRC2012 dataset [28]. For a fair evaluation, we composed TinyImageNet-A by collecting the intersected classes of TinyImageNet and ImageNet-A. Using TinyImageNet-A, we evaluated the robustness of four different networks: the backbone network trained with (1) the standard data augmentation, (2) MixUp, (3) CutMix, and (4) GridMix. All the different training schemes adopted the same backbone network.

The results are presented in Table 6. Although the accuracy of the network is extremely low in general, GridMix without or with Eq. (4) outperforms all the other methods. In particular, focusing on the relative gain, we observe that GridMix achieved an approxi-

Table 6
Comparison of robustness on the TinyImageNet-A dataset. “w. and “wo. mean with and without, respectively. The bold text indicates the best performance in comparison with the competitors; this indication is used throughout the paper.

Network	Top-k	Baseline	MixUp	CutMix	GridMix wo. Eq. (4)	GridMix w. Eq. (4)
VGG11BN	Top-1	2.29	2.29	2.16	2.02	**2.57**
	Top-5	8.28	8.82	8.38	8.35	**9.13**
VGG19BN	Top-1	2.12	2.98	2.87	2.46	**3.04**
	Top-5	9.27	10.74	10.19	9.75	**10.84**
ResNet50	Top-1	2.02	1.81	2.38	2.46	**2.47**
	Top-5	8.45	8.24	10.20	10.81	**10.97**
ResNext50	Top-1	2.12	2.98	2.87	2.46	**3.04**
(32x4d)	Top-5	7.90	6.39	10.11	**10.67**	9.89

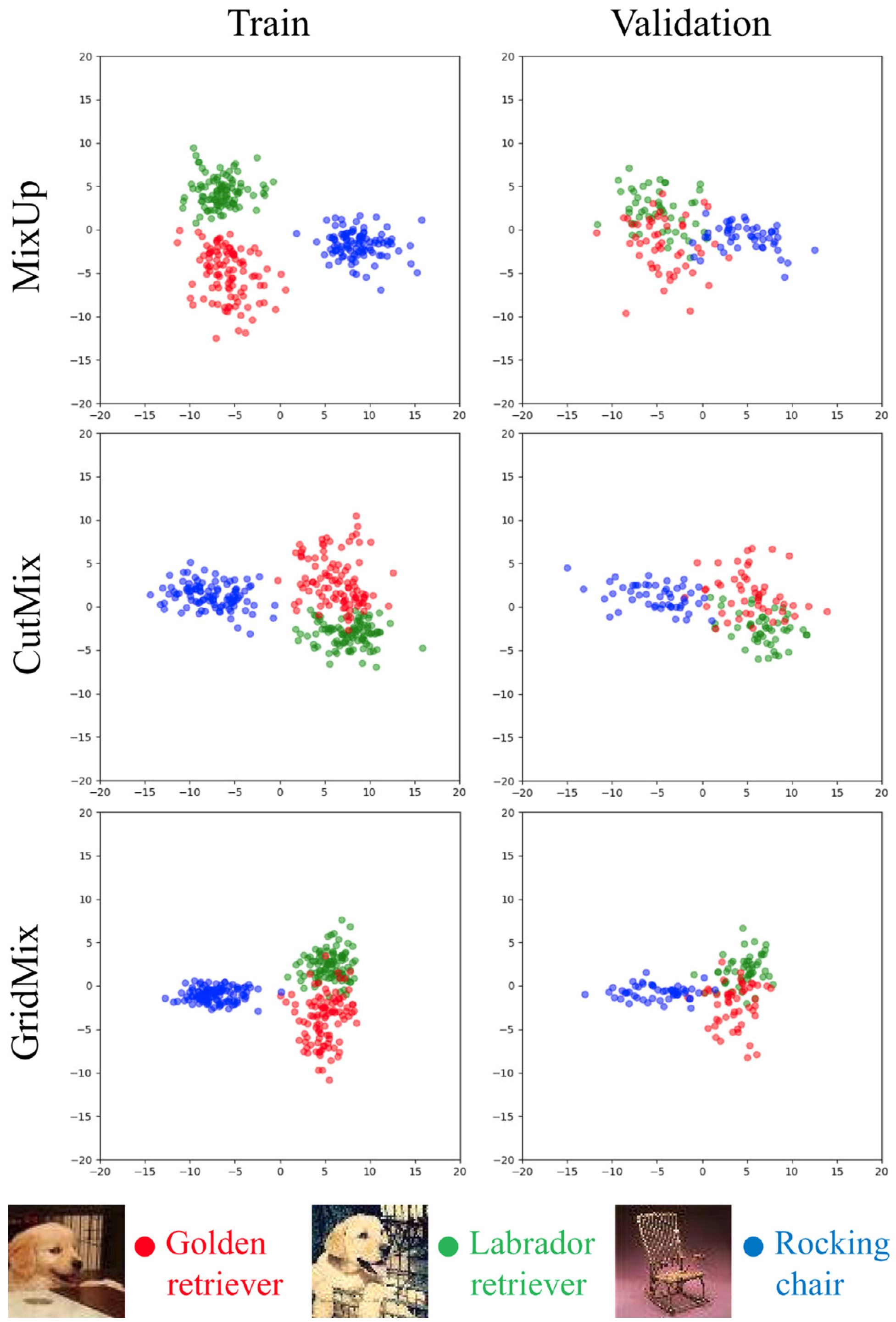


Fig. 3. Visualization of the penultimate layer's activations of MixUp, CutMix, and GridMix. We used VGG11 trained with TinyImageNet in this experiment. We observe that GridMix retains fairly similar activations between training and validation.

mate improvement of 10% over MixUp; MixUp delivers the second-best performance of all the methods. Through the results of the experiment using TinyImageNet-A, we confirm that GridMix clearly improves the robustness of the network to natural adversarial examples.

4.4. Effects of regularization techniques

To show the effects of different regularization techniques, we utilized the visualization of penultimate layer representations, as suggested in [31]. For visualizing penultimate layer representations, we (1) specified three classes, (2) computed two orthonormal bases for all the samples from the three classes, and (3) projected all the samples from the three classes to these two orthogonal bases. Of the three classes, two were semantically similar (Golden retriever and Labrador retriever), whereas the third was semantically fairly different (Rocking chair). Using this visualization, we focused on analyzing the density of the activation clusters and how effectively the semantic similarity is captured. Finally, we compared the activation distribution of training samples, as well as that of validation samples, and examined whether they share common characteristics.

In a comparison of the activation distributions from MixUp, CutMix, and GridMix, it can be seen that the clusters of GridMix are clearly denser than those of MixUp and CutMix. A tighter cluster implies that intra-class variations of the learned representation are smaller. This is a desirable property for improving classification. (In an ideal scenario, all samples in the same class should be at the same coordinate.) An additional interesting point to investigate was the activation distribution from semantically similar classes. For all three methods, the activations from similar classes are closer than those from other classes.

More importantly, we compared the distributions of training samples and validation samples for all three methods. This is because the goal of a regularization technique is to maintain the strong performance gain in the validation samples. For this reason, our interest lies in how closely the activation distribution from training samples matches that from validation samples. In Fig. 3, we observe that GridMix retains an activation distribution from the validation sample that is fairly similar to that from the training sample. In particular, GridMix can almost retain certain aspects such as the center of the cluster, the distance from other classes, and density. In the case of MixUp and CutMix, it can be observed that the distribution during training is not retained in the validation phase; for example, the distance between classes is reduced, and the center of the distribution is shifted.

5. Conclusion

In this paper, we proposed GridMix, a novel regularization method that utilizes a grid-based representation for augmentation and the label of a local patch as additional supervision. More specifically, the proposed method consists of two components: (1) data augmentation by means of grid-based image mixing (local data augmentation) and (2) a local context loss to predict patch-level labels (local context mapping). These two local constraints lead the network model to be locally overfitting, which naturally guides the network to consider the overall regions of an image, instead of focusing only on a few subregions (i.e. global overfitting).

The results of extensive experiments and evaluations indicate that GridMix achieves a new state-of-the-art performance in terms of classification and robustness and a comparable performance in terms of weakly supervised object localization. In case studies on the activation distribution of three classes and the visualization of attention maps, we clearly showed the effects of GridMix at intermediate activations. This explains why GridMix effectively alleviates overfitting, helping regularize the network.

The local context mapping of GridMix might suffer from the performance degradation when a cell only contains the background or non-targeted objects (uncorrelated with the class label). Currently, our GridMix utilizes the global context mapping to reduce such a degradation. As a future work, we consider to adaptively choose the cells that contain the target object and then to impose the local context mapping only for those selected cells. To do so, we plan to use the pseudo-label or the attention map derived from the network itself. We expect that this new approach will improve the prediction accuracy as it leads to focus on the local regions having relevant information.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the MSIP (NRF-2019R1A2C2006123, 2020R1A4A1016619), and by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (2020-0-01361, Artificial Intelligence Graduate School Program (YONSEI UNIVERSITY)). This work was also supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2016-0-00288) supervised by the IITP and by IITP grant funded by the Korea government (MSIP) (No. 2018-0-00198), Object information extraction and real-to-virtual mapping based AR technology.

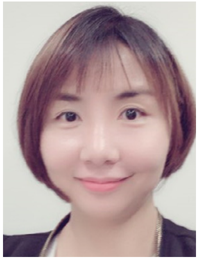
References

[1] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on CVPR, 2017, pp. 4700–4708.
[2] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE Conference on CVPR, 2018, pp. 7132–7141.
[3] X. Zhang, Y. Wei, J. Feng, Y. Yang, T.S. Huang, Adversarial complementary learning for weakly supervised object localization, in: Proceedings of the IEEE Conference on CVPR, 2018, pp. 1325–1334.
[4] J. Choe, H. Shim, Attention-based dropout layer for weakly supervised object localization, in: Proceedings of the IEEE Conference on CVPR, 2019, pp. 2219–2228.
[5] J. Lee, E. Kim, S. Lee, J. Lee, S. Yoon, Ficklenet: weakly and semi-supervised semantic image segmentation using stochastic inference, in: Proceedings of the IEEE Conference on CVPR, 2019, pp. 5267–5276.
[6] J. Peng, G. Estrada, M. Pedersoli, C. Desrosiers, Deep co-training for semi-supervised image segmentation, Pattern Recognit. 107 (2020) 107269.
[7] Y. Shi, Y. Han, Q. Zhang, X. Kuang, Adaptive iterative attack towards explainable adversarial robustness, Pattern Recognit. 105 (2020) 107309.
[8] J. Hang, K. Han, H. Chen, Y. Li, Ensemble adversarial black-box attacks against deep learning systems, Pattern Recognit. 101 (2020) 107184.
[9] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929–1958.
[10] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, C. Bregler, Efficient object localization using convolutional networks, in: Proceedings of the IEEE Conference on CVPR, 2015, pp. 648–656.
[11] K.K. Singh, Y.J. Lee, Hide-and-seek: forcing a network to be meticulous for weakly-supervised object and action localization, in: IEEE ICCV, IEEE, 2017, pp. 3544–3553.
[12] T. DeVries, G.W. Taylor, Improved regularization of convolutional neural networks with cutout, arXiv:1708.04552 (2017).
[13] H. Zhang, M. Cisse, Y.N. Dauphin, D. Lopez-Paz, Mixup: beyond empirical risk minimization, ICLR, 2018.
[14] S. Yun, D. Han, S.J. Oh, S. Chun, J. Choe, Y. Yoo, Cutmix: regularization strategy to train strong classifiers with localizable features, IEEE ICCV, 2019.
[15] A. Krizhevsky, G. Hinton, Learning multiple layers of features from tiny images, Department of Computer Science, University of Toronto, 2009 Master's thesis.
[16] S. CS231N, Tiny imagenet visual recognition challenge, 2017.
[17] J. Yu, D. Tao, M. Wang, Adaptive hypergraph learning and its application in image classification, IEEE Trans. Image Process. 21 (7) (2012) 3262–3272.
[18] S. Gidaris, P. Singh, N. Komodakis, Unsupervised representation learning by predicting image rotations, ICLR, 2018.
[19] T. Chen, X. Zhai, M. Ritter, M. Lucic, N. Houlsby, Self-supervised gans via auxiliary rotation loss, in: Proceedings of the IEEE Conference on CVPR, 2019, pp. 12154–12163.
[20] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, Y. Bengio, Manifold mixup: better representations by interpolating hidden states, in: Proceedings of the 36th International Conference on Machine Learning, 97, PMLR, 2019, pp. 6438–6447.
[21] J. Yu, Y. Rui, Y.Y. Tang, D. Tao, High-order distance-based multiview stochastic learning in image classification, IEEE Trans. Cybern. 44 (12) (2014) 2431–2442.
[22] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: ICLR, 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings, 2015.
[23] S. Ioffe, C. Szegedy, Batch normalization: accelerating deep network training by reducing internal covariate shift, in: Proceedings of ICML, in: Proceedings of Machine Learning Research, 37, PMLR, 2015, pp. 448–456.
[24] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on CVPR, 2016, pp. 770–778.
[25] K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, in: ECCV, 2016, pp. 630–645.
[26] S. Zagoruyko, N. Komodakis, Wide residual networks, in: Proceedings of BMVC, BMVA Press, 2016, pp. 87.1–87.12.

[27] S. Xie, R. Girshick, P. Dollár, Z. Tu, K. He, Aggregated residual transformations for deep neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1492–1500.
[28] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, Ieee, 2009, pp. 248–255.
[29] C. Wah, S. Branson, P. Welinder, P. Perona, S. Belongie, The caltech-ucsd birds-200-2011 dataset (2011).
[30] J. Gilmer, D. Hendrycks, A discussion of 'adversarial examples are not bugs, they are features': adversarial example researchers need to expand what is meant by 'robustness', Distill 4 (8) (2019) e00019–1.
[31] R. Müller, S. Kornblith, G.E. Hinton, When does label smoothing help? in: Advances in Neural Information Processing Systems, 2019, pp. 4694–4703.
[32] S. Pu, Y. Song, C. Ma, H. Zhang, M.-H. Yang, Deep attentive tracking via reciprocative learning, in: Advances in Neural Information Processing Systems, 2018, pp. 1931–1941.
[33] S. Zagoruyko, N. Komodakis, Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer, ICLR, 2017.

Kyungjune Baek is currently a Ph.D. student at the School of Integrated Technology, Yonsei University, South Korea. He received his B.S. degree in electrical and electronic engineering, and computer science also from this university. His research interests are in the areas of computer vision and deep learning, especially generative adversarial networks, self supervised learning, and image-to-image translation.

Duhyeon Bang received his B.S. degree in biomedical engineering in 2013 and his M.S. and Ph.D. degrees at the School of Integrated Technology from Yonsei University, South Korea in 2016 and 2020, respectively. He is currently a research scientist with SKTBrain, SKTelecom, Seoul, South Korea. His research interests are in the areas of computer vision and deep learning, especially generative adversarial networks, knowledge distillation, and image classification.



Hyunjung Shim received her B.S. degree in electrical engineering from Yonsei University, Seoul, Korea, in 2002, and her M.S. and Ph.D. degrees in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, USA, in 2004 and 2008, respectively. She was with Samsung Advanced Institute of Technology, Samsung Electronics Company, Ltd., Suwon, Korea, from 2008 to 2013. She is currently an assistant Professor with the School of Integrated Technology, Yonsei University. Her research interests include generative models, deep neural networks, classification/recognition algorithms, 3-D vision, inverse rendering, face modeling, and medical image analysis.