Signal, Image and Video Processing (2023) 17:1245–1253
https://doi.org/10.1007/s11760-022-02332-x

ORIGINAL PAPER

LMix: regularization strategy for convolutional neural networks

Linyu Yan[1] · Kunpeng Zheng[1] · Jinyao Xia[1] · Ke Li[1] · Hefei Ling[2]

Received: 6 May 2022 / Revised: 12 July 2022 / Accepted: 28 July 2022 / Published online: 21 August 2022

Abstract
Deep convolutional neural networks perform well in the field of computer vision, but exhibit undesirable behaviors such as memorization and sensitivity to adversarial examples. Therefore, proper regularization strategies are needed to alleviate these problems. Currently, regularization strategies with mixed sample data augmentation perform very well, and these algorithms allow the network to generalize better, improve the baseline performance of the model. However, interpolation-based mixed sample data augmentation distorts the data distribution, while masking-based mixed sample data augmentation results in excessive information loss for overly regular shapes of masks. Although mixed sample data augmentation is proven to be an effective method to improve the baseline performance, generalization ability and robustness of deep convolutional models, there is still room for improvement in terms of maintaining the of image local consistency and image data distribution. In this paper, we propose a new mixed sample data augmentation-LMix, which uses random masking to increase the number of masks in the image to maintain the data distribution, and high-frequency filtering to sharpen the image to highlight recognition regions. We applied the method to train CIFAR-10, CIFAR-100, SVHN, and Tiny-ImageNet datasets under the PreAct-ResNet18 model to evaluate the method, and obtained the latest results of 96.32, 79.85, 97.01, and 64.16%, respectively, which are 1.70, 4.73, and 8.06% higher than the optimal baseline accuracy. The LMix algorithm improves the generalization ability of the state-of-the-art neural network architecture and enhances the robustness to adversarial samples.

Keywords Mixup · Data augmentation · Deep convolutional neural networks · Regularization strategies

1 Introduction

Deep convolutional neural networks have shine in various computer vision tasks, such as image classification [1,2,5], object detection [29,30], anomaly detection [33,34], semantic segmentation [14], and image super-resolution [15]. Deep convolutional neural networks follow the empirical risk minimization principle [16] to minimize the average error when performing training. Also, when the deep convolutional neural network is used to extract features from an input image, the larger the training sample, the greater the learning effect and generalization capacity of the model. For instance, the network of Foret et al. [1] was modeled using the JFT-300M dataset [31] with 4.8 billion parameters. Mahajan et al. [2] used the ImageNet-22k dataset to model their network, which has 8.2 billion parameters. Tan et al [3]. used the ImageNet-22k dataset to model their network, which has 1.2 billion parameters. To further improve the training accuracy and speed, many scholars have proposed some training strategies, such as regularization techniques, data augmentation strategies [6,8–10], etc. The regularization technique prevents overfitting in networks with more parameters than input data, as well as algorithmic generalization by avoiding training coefficients of perfect-fit data samples. Data augmentation can prevent model overfitting and increase the number of samples to improve model generalization, mainly including geometric space change, pixel color transformation, and multiple sample fusion.

Currently, mixed sample data augmentation [8–12] techniques based on Vicinal Risk Minimisation [13,36] have obtained good results in a variety of applications, particu-

✉ Kunpeng Zheng
zhengfreeking@163.com

Linyu Yan
361953203@qq.com

Jinyao Xia
2464274209@qq.com

Ke Li
1028132487@qq.com

[1] Hubei University of Technology, Wuhan, Hubei, China

[2] Huazhong University of Science and Technology, Wuhan, Hubei, China

larly classification tasks. Lu et al. [35] employs CutOut [6] and Mixup [8] to alleviate the impact of the above problems, improving the performance of the model. The Vicinal Risk Minimisation based data augmentation approach extracts additional dummy samples from the training samples to boost support for the training distribution. This also leads to the goal of expanding samples to increase data space without distorting the data distribution, nevertheless, larger samples unavoidably have distorted data distribution [13]. To ensure that the data enhancement strategy can produce good results for the network, the following characteristics should be maintained: the virtual samples and the real samples should have a good acquaintance; the data augmentation strategy can improve the model's generalization ability; and the data augmentation strategy can improve the model's robustness against noise.

Mixed sample data augmentation is the modification of sample data to build an extended dataset for training models. Mixed sample data augmentation proposed so far is broadly classified into two types: interpolation and masking. Mixed sample data augmentation for Interpolation has Mixup [8], which is a mixed sample data augmentation based on the idea of Vicinal Risk Minimisation, and Mixup suggests a general vicinal distribution, the mixed distribution, as illustrated in Fig. 1a. Mixed sample data augmentation for masking has CutMix [9], which proposes patches are cut and pasted among training images where the ground truth labels are also mixed proportionally to the area of the patches, as illustrated in Fig. 1b. Both strategies improve the baseline performance of the deep convolutional model. In terms of picture data distribution, CutMix trumps Mixup.

In this paper, we propose a new mixed sample data augmentation LMix, as illustrated in Fig. 1c. The main ideas are as follows: (1) use random masking to increase the number of image masks while effectively ensuring the local consistency of the image. (2) use high frequency filtering to sharpen the image to highlight the recognition area. The rest of this paper is organized as follows. In Sect. 2, we review the existing work on data enhancement strategies. Then, we present the implementation of the LMix algorithm in Sect. 3. In Sect. 4, we conduct a large number of experiments to demonstrate the effectiveness and efficiency of the proposed algorithm. Finally, we conclude in Sect. 5.

2 Related work

Data augmentation With the deepening of the deep network, the required learning parameters continue to increase, which inevitably leads to overfitting. When the dataset is too small, too many parameters can fit all the characteristics of the dataset rather than the commonalities between the data [25,26]. Data augmentation generates virtual samples from real samples to expand the dataset size, which can alleviate the problem of model overfitting and make the training data as close as possible to the test data, thus improving the accuracy. At the same time, data augmentation can force the model to improve robustness and make the model more generalizable. Early data augmentation algorithms were transformations of images using geometric transformations including flip, rotate, crop, distort, scale, etc., and color transformations including noise, blur, color transformation, erase, fill, etc. Lopes et al. [4] added Gaussian blocks to Cutout to make the model more stable without losing model accuracy by adding noise to randomly selected blocks in the input image. Also, this method can be used in combination with other regularization methods and data enhancement strategies. He et al. [5] trained the deep residual network with random left-right flipping and cropping of the image data to improve the generalization ability of the model. This allowed the data samples to be expanded and greatly improved the generalization ability of the model. DeVries et al. [6] proposed masking regularization, a data augmentation approach comparable to random erasure. They apply random masking on the image, masking it with a fixed-size rectangle. Within the rectangle, all values are set to 0 or other solid color values, and the erased rectangular section may or may not be totally in the picture. Taylor and Nitschke [7] analyzed the effectiveness of geometric and photometric (color space) transformations. They analyzed geometric changes such as flipping, as well as color space transformations such as color dithering (random color manipulation), edge improvement, and principal component analysis. Simply conducting simple image processing on individual photographs might lead to a slew of issues. For instance, operations such as flip, shear, and rotate are not safe for the dataset [27], while the color transformation enhancement approach is biased from a color space perspective with more diversity of color variations, resulting in insufficient enhancement and poor learning and underfitting of the color space, while the transformation is unsafe.

Mixup The mixed sample data augmentation not only has good generalization ability, but also has excellent robustness, both for data containing noisy labels and against sample attacks. The fused images obtained by the mixed sample data augmentation are difficult to understand under the human perspective, yet the experimental results are excellent. To make the model baseline performance better, several data augmentation algorithms have been proposed by the sample fusion approach to enhance the accuracy and generalization of the model. Zhang et al. [8] proposed Mixup (Fig. 1a) one such data-dependent regularizer, synthesizes additional training examples by interpolating random pairs of inputs x_i, x_j and their corresponding labels y_i, y_j as:

$$\hat{x} = \lambda x_i + (1 - \lambda) x_j \tag{1}$$

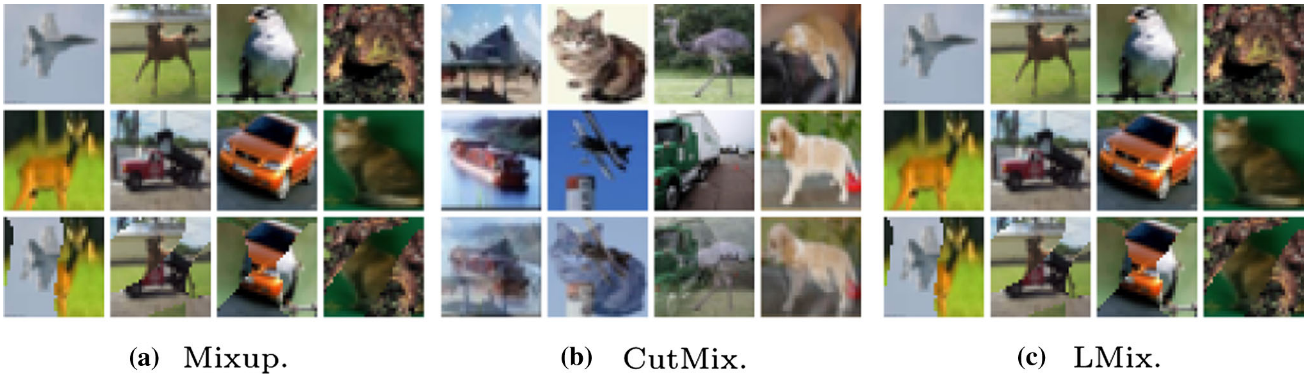

Fig. 1 Generated images of CutMix, LMix, and Mixup on the CIFAR dataset

$$\hat{y} = \lambda y_i + (1 - \lambda) y_j \quad (2)$$

where $\lambda \in [0, 1]$ is sampled from a Beta distribution such that $\lambda \sim \text{Beta}(\alpha, \alpha)$ and $(\hat{x}, \hat{y})$ is the new example. Mixup for training the model in a convex combination of sample and label pairs. It enhances the linear representation of the training data, models samples from different domains, and improves the generalization performance of the model. Image enhancement algorithms are simple and data independent. However, Mixup methods distort the data distribution of the images, while generating virtual samples that are not very interpretable. Yun et al. [9] proposed Cutmix, a mixed sample data augmentation for masking. As shown in Fig. 2, Cutmix implants an input random rectangular region into another rectangular region. However, using a regular cropping approach can cause the image to lose a lot of information. Vera et al. [10] proposed an extension of input data blending to blending the output of the intermediate hidden layer. Making the network transformed to the input data is smoother and more uniform, which leads to improved model performance and generalization. Kim et al. [11] proposed a blending method based on saliency and local statistics of the given data. They added significance analysis to CutMix. Harris et al. [12] proposed an improved method based on CutMix. They verified that the hybrid method of clipping is more advantageous than the hybrid method of interpolation in preserving the image data distribution, and designed an irregular mask to mask the image when the spatial size of the data sample is increased. The spatial scale increases.

3 Method

We discover that masked mixed sample data augmentation is more effective than interpolated mixed sample data augmentation in preserving data distribution, especially on convolutional neural networks. Convolutional neural networks are locally consistent, means that each neuron is only linked to one portion of the input neuron at a specified geographical position. Neurons are locally linked in the spatial dimension but completely connected in the depth dimension in picture convolution procedures. For the two-dimensional image itself, the local pixel correlation is also strong. This local connectedness guarantees that the learned filter responds to the local input characteristics as strongly as possible. It is extremely critical for neural networks to successfully preserve the pictures' local consistency. Meanwhile, the disadvantage of the interpolative mixed sample data augmentation is that it only uses regular masking to operate the image, and the number of masks cannot be well guaranteed, thus we must increase the number of masks while keeping local consistency.

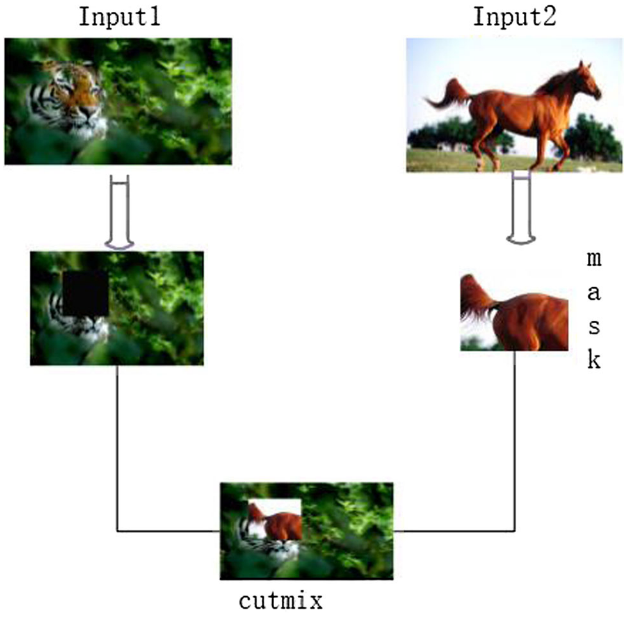

Fig. 2 The process of generating CutMix

In this section, we propose LMix, mixed sample data augmentation that provides the greatest results in terms of local consistency and the number of masks in the picture, as shown in Fig. 3. LMix employs an masking mixed sample data augmentation to preserve the image's local consistency.

Its algorithm is implemented as:

Let $x \in R^{W \times H \times C}$ denote a training set, y denote the training set's label, and (x_A, y_A) and (x_B, y_B) represent two feature target vectors chosen at random from the training data. LMix's purpose is to create a new sample $(\hat{x}, \hat{y})$ by merging two training samples (x_A, y_A) and(x_B, y_B). The resulting training sample$(\hat{x}, \hat{y})$ is utilized to train the original loss function-trained model. It is defined as follows.

$$\hat{x} = \text{mask} \cdot x_A + (1 - \text{mask}) \cdot x_B \tag{3}$$

$$\hat{y} = \lambda y_A + (1 - \lambda) y_B \tag{4}$$

where mask $\in \{0, 1\}^{W \times H}$ is the binary mask, which refers to the bits deleted and filled from the two pictures, and '1' represents the binary mask filled with 1. As in Mixup [8], the combined ratio λ between the two data points is obeying the Beta(α, α) distribution, where Beta() indicates Beta function, and $\alpha \in (0, \infty)$. Compared to Cutmix [9], which directly intercepts a regular patch from a image to replace the image region of the target image, we use a mask made by combining a single region and its adjacent regions, which can reduce the number of binary mask conversions. To obtain the binary mask, we first apply the two-dimensional discrete Fourier transform to the image to convert it from the spatial domain to the frequency domain, and then obtain the low-frequency image of the image, the high-frequency component of the image through the low-frequency component, and the high-frequency filtered and enhanced image of the image through the high-frequency component. We define Z as a complex random variable with a value domain of $Z = C^{H \times W}$ and densities of $P_{R(Z)} = N(0, I_{W \times H})$ and $P_{I(Z)} = N(0, I_{W \times H})$, and $N(0, I)$ denotes multivariate Gaussian distribution. The real and imaginary components of the input are denoted by $R(Z)$ and $I(Z)$, respectively. By attenuating the high-frequency component of Z, a low-pass filter is created. By attenuating the high-frequency component of Z, we may create a low-pass filter.

$$f_{\text{LP}}(z, \delta)[i, j] = \frac{z[i, j]}{\text{freq}(w, h)[i, j]^{\delta}} \tag{5}$$

f_{LP} denotes low-pass filter, Let freq$(w, h)[i, j]$ denote the magnitude of the sample frequency corresponding to the i, jth bin of the $w \times h$ discrete Fourier transform, $z[i, j]$ denote a complex random variable with a value domain of $z = C^{i \times j}$, δ denote decay power, The image high-pass filter is obtained by the obtained low-pass filter.

$$f_{\text{HP}}(Z, \delta) = 1 - f_{\text{LP}}(Z, \delta) \tag{6}$$

f_{HP} denotes high-pass filter, The sharpened image is then obtained by passing it through a high-pass filter.

$$g_{\text{mask}}(x, y) = f^{-1}\{[1 + k \cdot f_{\text{HP}}(Z, \delta)]F(u, v)\} \tag{7}$$

where $g_{\text{mask}}(x, y)$ denotes the high-frequency filtered enhanced image, $F(u, v)$ denotes the Fourier variation of the original image, $f_{\text{HP}}(Z, \delta)$ denotes the high-pass filter, δ is the given attenuation frequency, and f^{-1} denotes the discrete Fourier inverse transform. Finally obtaining the sampled binary mask mask now all that remains is to convert the grayscale image to a binary mask such that the average is some given λ. Let top(n, x) return a set containing the top n elements of the input x. Setting the value of the top λ, w, h elements of some grayscale image g to '1' and the value of all other elements to '0', we obtain a binary mask with an average λ.

$$\text{mask}\,(\lambda, g)\,[i, j] = \begin{cases} 1, \text{ if } g_{[i,j]} \in \text{top}\,(\lambda w h, g) \\ 0, \text{ otherwise} \end{cases} \tag{8}$$

We first sample a random complex tensor whose real and imaginary components are both independent and Gaussian distributed. Then, each component is scaled according to its frequency by the parameter δ, such that higher δ values correspond to increased attenuation of high-frequency information. Next, the Fourier inverse transform of the complex tensor is performed and its real part is taken to obtain a grayscale image. Finally, the top scale of the image is set to '1' and the rest of the scale to '0' to obtain the binary mask.

4 Experiment

In this section, we apply LMix to ResNet [19], DenseNet [21], and WideResNet [20] models on the CIFAR-10, CIFAR-100 [17], Fashion-MNIST, SVHN, and Tiny-ImageNet [18] datasets for image classification tasks to evaluate the enhancements and generalization improvements to the model baseline that LMix can provide. The same hyperparameters are used for all models in order to fairly compare and evaluate the performance improvement of different mixed-sample data augmentation on generalization and augmented baselines. In addition, the parameters of all mixed sample data augmentation algorithms are selected to produce the best results in the corresponding papers. We replicate all studies when possible and publish the average performance and standard deviation following the last phase of training. In all tables, we highlight the best outcomes and those that are within their margin of error.

Table 1 The accuracy of the approach for the image classification task in CIFAR-10 using the PreAct-ResNet-18, WideResNet-28-10, DenseNet-BC-190 (Dense), and PyramidNet-272-200 models

Data set	Model	Baseline	FMix	Mixup	CutMix	LMix
CIFAR-10	ResNet-18	94.62	96.14	95.67	95.97	96.32
	WRN	95.32	96.41	96.69	96.63	96.58
	Dense	96.32	97.30	97.01	96.95	97.36
	Pyrmid	97.31	98.54	97.90	98.18	98.58

Table 2 Accuracy of the algorithm using PreAct-ResNet-18,PreAct-ResNet-34, WideResNet-28-10, and DenseNet-BC-190 models to test the algorithm for the image classification task in CIFAR-100

Data set	Model	Baseline	FMix	Mixup	CutMix	LMix
CIFAR-100	ResNet-18	75.12	79.65	77.34	78.48	79.85
	ResNet-34	76.58	81.55	78.54	79.56	81.86
	Dense	78.24	82.03	81.95	81.84	81.91
	Pyrmid	81.64	83.75	83.23	82.69	83.95

4.1 Image classification

4.1.1 CIFAR classification

This section first discusses the results of the image classification task on the CIFAR 10/100 dataset. On the CIFAR dataset we train: the PreAct-ResNet18 citeref19, WideResNet-28-10 citeref20, DenseNet-BC-190 citeref21, and PyramidNet-272-200 [22] models. We found that the regularization methods including cutout [6], Mixup [8], CutMix citeref9, and FMix [12] need a longer training time to reach convergence. As a result, we set the epoch of all models to 300, the initial learning rate to 0.1, and decay at 75, 150, and 225 epochs in multiples of 0.1, with a batch size of 128. Table 1 compares the performance of the approach to that of other cutting-edge data augmentation and regularization methods. All trials were repeated five times, and the best performance during training is presented as the average.

Hyperparameter setting We set the hyperparameter α of LMix to 1 and the decay rate δ to 3. Set the cropping area of Cutmix [9] and Cutout [6] to 16×16. For Mixup, we set the hyperparameter α to 1, set the hyperparameter α and decay rate δ of FMix [12] to 1 and 3, and the hyperparameters α of Patchup [10],$Patchup_{prob}$,x, and block size are set to 2, 0.7, 0.5, and 7, respectively.

LMix is applicable to a variety of models As shown in Table 1, LMix applies to various convolutional neural networks, while LMix significantly improves the baseline performance of various lightweight models, and for the ResNet-18 [19] model, LMix improves the most accuracy over the baseline performance by 1.51% and on the average accuracy over the baseline performance by 1.68%. For the WideResNet-28 [20] model, LMix improves 1.23% over the maximum accuracy of the baseline performance and 1.29% over the average accuracy of the baseline performance. For the DenseNet [21]

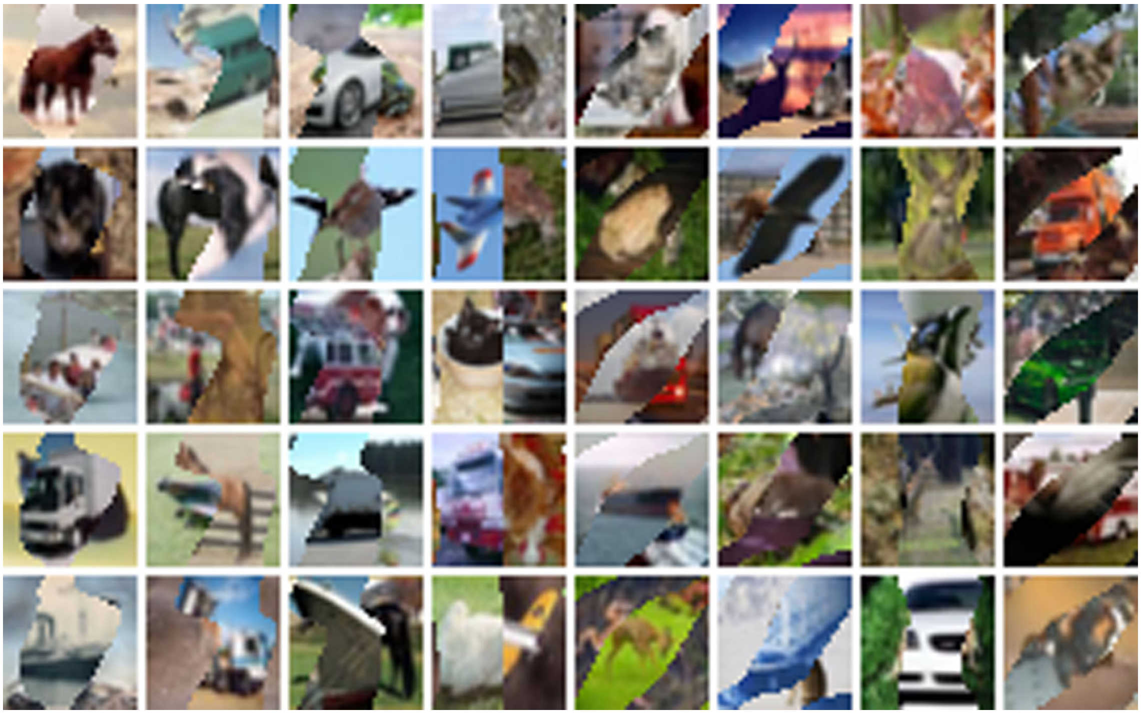

Fig. 3 Virtual sample of sample fusion acquired from CIFAR-100

Table 3 Accuracy of the image classification task using PreAct-ResNet-18 test algorithm in Tiny-ImageNet

Model	MaxAcc (%)	Acc (%)
Baseline	55.94	55.86
+CutMix	64.08	63.84
+FMix	63.33	62.23
+Mixup	61.96	61.89
+LMix	64.16	63.92

model, LMix improved the maximum accuracy over baseline performance by 1.05% and the average accuracy over baseline performance by 1.11%. For the Pyramid [22] model LMix improved the maximum accuracy over the baseline performance by 1.32% and the average accuracy over the baseline performance by 1.33%.

*LMix performance on CIFAR-10/100*The results in Table 2 show that the same models were trained on the CIFAR-10 dataset, and LMix provided significant improvements over the other hybrid sample enhancement algorithms. For ResNet-18, LMix outperforms cutout by 1.16%, Mixup by 0.72%, Cutmix by 0.42%, FMix by 0.29%, and patchup by 0.62% in terms of accuracy for the image classification task. LMix also performs very well on the CIFAR-100 dataset, as shown in Table 2. For ResNet-18, LMix outperformed the baseline by 4.73%, outperformed FMix by 0.2%, outperformed CutMix by 0.37%, and outperformed Mixup by 2.51% on the image classification task.

The results obtained in Fig. 4 indicate that LMix has the highest accuracy for the image classification task trained with ResNet-18 on CIFAR-100 with hyperparameter $\alpha = 1$, while outperforming Mixup, CutMix, and FMix. we have explored the performance of LMix for ResNet-18 and ResNet-34 on CIFAR-100. As shown in Fig. 4, we found an improvement in accuracy for both classification tasks.

4.1.2 Tiny-ImageNet

We trained the PreAct-ResNet18 network on the Tiny-ImageNet [18] dataset, which contains 200 classes with 500 training images and 50 test images per class with a resolution of 64×64. We trained the model with an initial learning rate of 0.1 for 200 epochs, and we used a decay learning rate of 0.1 at 150 and 180 epochs. we set the momentum to 0.9. In the case of Mixup weights λ, for the Mixup, we set $\alpha = 1$ as described in the Mixup. For CutMix, we chose $\alpha = 1$, which is the best performance in [0.2, 0.5, 1.0], while for FMix, we chose $\alpha = 1.0$, for Cutout and CutMix with a cropping region of 16×16.

In the experiments using the Tiny-ImageNet dataset, compared with other hybrid baselines, LMix showed significant improvements in generalization performance and improved model accuracy (Table 3). With the same number of epochs trained, LMix achieves an accuracy of 64.06%, which is 0.08% higher than the strongest baseline.

Table 4 Training PreAct-ResNet18 on the Fashion-MNIST dataset to evaluate LMix

Model	MaxAcc (%)	Acc (%)
Baseline	95.70	95.52
+CutMix	96.02	95.93
+Mixup	96.26	96.20
+LMix	96.64	96.62

Table 5 Accuracy of the algorithm using PreAct-ResNet-18, PreAct-ResNet-34, WideResNet-28-10, and DenseNet-BC-190 models to test the algorithm for the image classification task in CIFAR-100

Data set	Model	Baseline	Mixup	CutMix	LMix
SVHN	ResNet-18	96.53	96.63	96.57	97.01
	ResNet-34	97.04	97.21	97.44	97.66
	WRN	97.28	97.48	97.69	97.73

4.1.3 Fashion-MNIST

We train the PreAct-ResNet18 network on the Fashion-MNIST dataset, a fashion product dataset containing 70,000 28×28 grayscale images in 10 categories with 7000 images in each category. The training set has 60,000 images and the test set has 10,000 images. Fashion MNIST shares the same image size, data format, and training and test splitting structure as the original MNIST. We trained the PreAct-ResNet18 [19] model, where we trained the model with an initial learning rate of 0.1 for 200 epochs, and we used a decay learning rate of 0.1 at 150 and 180 epochs. we set the momentum to 0.9. in the case of Mixup weights λ, for Mixup, we set $\alpha = 1$ in Mixup. Set the cropping area for Cutout and CutMix to 16×16.

In the experiments using the Fashion-MNIST dataset, compared with other hybrid baselines, LMix showed significant improvements in generalization performance and improved model accuracy (Table 4). With the same number of epochs trained, LMix achieves an accuracy of 96.62%, which is 1.1% higher than the strongest baseline.

4.1.4 SVHN

We train multiple image classification network models on the SVHN dataset, a numerical classification benchmark dataset containing 600,000 32×32 RGB images of printed digits (from 0 to 9) cropped from door sign images. The cropped images are centered on the digit of interest, but nearby digits and other distractors are retained in the images. SVHN has three sets: a training set, a test set, and an additional set con-

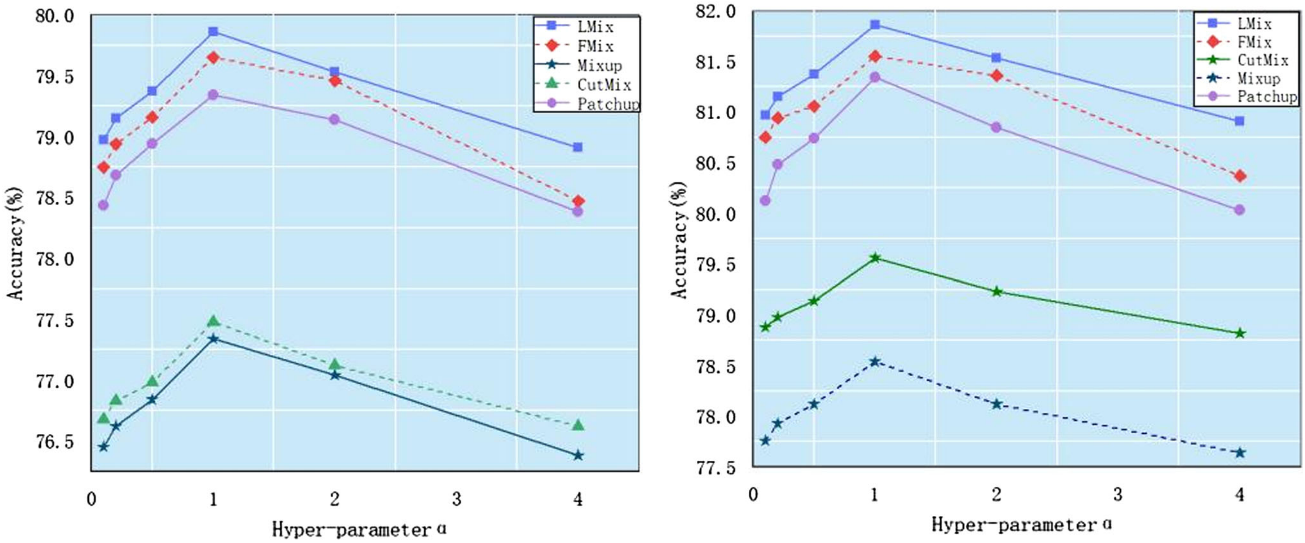

Fig. 4 Effect of varying the value of hyperparameter α on the baseline accuracy of various algorithms under the CIRAF-100 data set

taining 530,000 images that are less difficult and can be used to aid in the training process. To evaluate the effect of LMix on the SVHN dataset, we applied LMix on PreAct-ResNet18, PreAct-ResNet34 and WideResNet-28-10, respectively. We set the epoch of the model to 300, the initial learning rate to 0.1, and decay at epochs of 75, 150, and 225 in multiples of 0.1, and set the batch size to 128. Also, we repeated the experiments several times to obtain the most reliable results.

LMix performance in SVHN: The results in Table 5 show that the same model is trained on the SVHN dataset and LMix provides significant improvements over other mixed-sample enhancement algorithms. For ResNet-18, LMix provides 0.48% higher accuracy than Mixup and 0.44% higher than Cutmix for the image classification task. Also when ResNet-34 and WideResNet-28-10 are applied, there is a good improvement in the accuracy and generalization of the model.

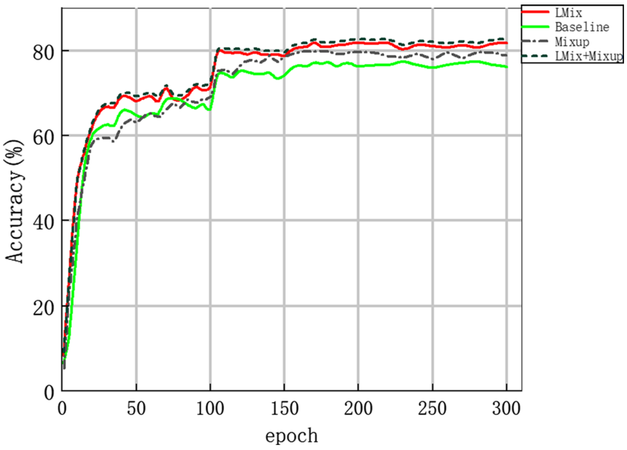

Fig. 5 Training PreAcResNet-34 on the CIFAR-100 dataset

4.2 Combining MSDAs

We trained the PreAct-ResNet-18 network on the CIFAR-100 dataset and used it to evaluate the effect of the algorithm combination. We train 300 epoch models with an initial learning rate of 0.1, and we use a decay learning rate of 0.1 at 100, 150, and 225 epochs, with batch size set to 128. for Mixup, we set the hyperparameter α to 1. We also set the hyperparameters α and δ of LMix to 1 and 3, respectively. we set the hyperparameters α and $delta$ of LMix+ The hyperparameter α is set to 1 for the Mixup combination.

As shown in Fig. 5, the accuracy of LMix for the image classification task with the PreAct-ResNet-34 model trained under the CIFAR-100 dataset is significantly higher than the baseline performance of Mixup and the original model, while the combined approach further improves the accuracy of the model after combining LMix with Mixup.

4.3 Robustness

When performing image classification tasks, the neural network is first trained and then minimized with respect to the error on the training sample, This learning rule is called empirical risk minimization. When small changes occur in the data samples it can have a significant impact on the performance of the model. For neural networks, most current neural networks set the model in a linear form to obtain faster computation speed, resulting in a very weak fight against perturbed samples. Certain data-dependent regularization techniques can mitigate the vulnerability of adversarial examples by interpolating the data to train the model. To evaluate the robustness of LMix against adversarial attacks, we

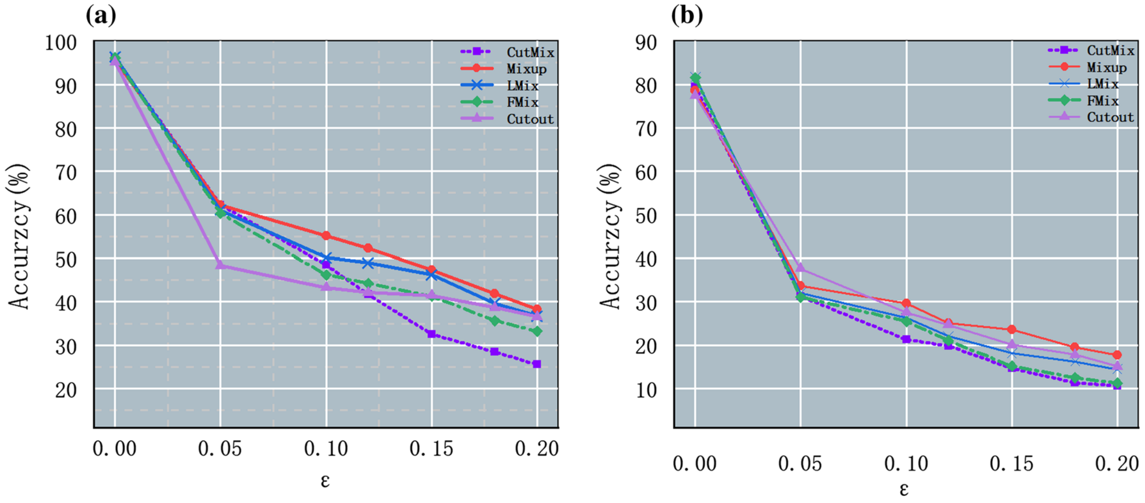

Fig. 6 Robust FGSM attack, **a** comparison at CIFAR-10 using the PreAc-tResNet-18 model, **b** comparison at CIFAR-100 using the PreAct-ResNet-34 model

compared the performance of PreActResNet18, PreActResNet34 on CIFAR-10 and CIFAR-100 with the adversarial examples generated by the FGSM attack described in.

Figure 6 shows the comparison of the impact of the state-of-the-art regularization techniques on the robustness of the model against FGSM attacks. Based on the results, we can see that LMix achieves the second-best performance in terms of robustness against adversarial attacks compared to other regularization methods. Our experiments show that in most cases, LMix is effective against attacks (Fig. 6).

5 Conclusion

In this paper, we propose LMix, mixed sample data augmentation which improves the classification performance and generalization ability of a model. The model is improved by preserving the local consistency of the image and then maximizing the number of masks, using a random masking approach to increase the number of masks on the image, and a high-frequency filter to sharpen the image to highlight recognition regions. We run a series of analyses to ensure the feasibility of the idea and then design preliminary experiments and find that LMix performs very well on the classification task. Applying LMix to the PreActResNet18 model to train the CIFAR-10 dataset yielded results that were 1.70% above the baseline, with an optimal result of 96.32%. Then we applied LMix to WideResNet-28 in the CIFAR-10 classification task, and PyramidNet could improve the highest baseline accuracy by 1.26 and 1.27%, respectively. For CIFAR-100, LMix significantly improves the baseline performance by 4.73%. We conducted experiments on the datasets SVHN, Tiny-ImageNet, and Fashion MINIST, respectively, which were 0.48% higher than the baseline accuracy on SVHN, 1.10% higher than the baseline accuracy on Fashion MINIST, and 8.06% higher than the baseline accuracy on Tiny-ImageNet. Finally, we conducted robustness experiments on LMix. Our experimental results show that LMix has excellent performance in terms of generalization performance and robustness against interference.

Author contributions LY and KZ completed the main manuscript text and experiments. JX prepared Table 4. KL prepared Table 5. All authors reviewed the manuscript.

Funding This work is funded by the National Natural Science Foundation of China under Grant No. 61772180, the Key R D plan of Hubei Province (2020BHB004, 2020BAB012).

Availability of data and materials All of our data sets come from public data sets.You can go to the corresponding official website to download.

Declarations

Conflict of interest Not applicable.

Ethical approval Not applicable.

References

1. P. Foret, A. Kleiner, H. Mobahi, B. Neyshabur, Sharpness-aware minimization for efficiently improving generalization. (2021), arXiv:2010.01412
2. D.K. Mahajan, R.B. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe, L.V. Maaten, Exploring the limits of weakly supervised pretraining. (2018), arXiv:1805.00932
3. M. Tan, Q. Le, EfficientNetV2: smaller models and faster training. (2021), ArXiv:2104.00298
4. R.G. Lopes, D. Yin, B. Poole, J. Gilmer, E.D. Cubuk, Improving Robustness without sacrificing accuracy with patch Gaussian augmentation. (2019), arXiv:1906.02611

5. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition. 2016 IEEE conference on computer vision and pattern recognition (CVPR), 770-778 (2016) arXiv:1512.03385
6. T. Devries, G.W. Taylor, Improved regularization of convolutional neural networks with Cutout. (2017), arXiv:1708.04552
7. L. Taylor, G.S. Nitschke, Improving deep learning using generic data augmentation. (2017), arXiv:Learning
8. H. Zhang, M. Cissé, Y. Dauphin, D. Lopez-Paz, Mixup: beyond empirical risk minimization. (2018), arXiv:1710.09412
9. S. Yun, D. Han, S. Oh, S. Chun, J. Choe, Y.J. Yoo, CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features. 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 6022-6031(2019) arXiv:1905.04899
10. V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, Y. Bengio, Manifold Mixup: Better Representations by Interpolating Hidden States. ICML. (2019) arXiv:1806.05236
11. J. Kim, W. Choo, H.O. Song, Puzzle Mix: Exploiting Saliency and Local Statistics for Optimal Mixup,(2020), arXiv:2009.06962
12. E. Harris, A. Marcu, M. Painter, M. Niranjan, A. Prügel-Bennett, J.S. Hare, FMix: Enhancing Mixed Sample Data Augmentation. (2020) arXiv:2002.12047
13. Chapelle, O., Weston, J., Bottou, L., apnik,V.: Vicinal risk minimization, in Advances in neural information processing systems. 416–422 (2001)
14. Y. Yuan, X. Chen, J. Wang, Object-contextual representations for semantic segmentation. (2020), arXiv:1909.11065
15. Dong, C., Loy, C.C., He, K., Tang, X.: Image Super-Resolution Using Deep Convolutional Networks. IEEE Trans. Pattern Anal. Mach. Intell. **38**, 295–307 (2016)
16. Vapnik, V.N.: The Nature of Statistic Learning Theory. (2000). https://doi.org/10.1007/978-1-4757-2440-0
17. Krizhevsky, A. , Hinton, G.P.: Learning multiple layers of features from tiny images. Handbook of Systemic Autoimmune Diseases **1**(4), (2009)
18. Le, Y., Yang, X.S.: Tiny ImageNet visual recognition challenge. 529 CSN 231N 7, 3 (2015). https://tiny-imagenet.herokuapp.com
19. K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks. (2016), arXiv:1603.05027
20. S. Zagoruyko, N. Komodakis, Wide residual networks. (2016), arXiv:1605.07146
21. G. Huang, Z. Liu, K.Q. Weinberger, Densely connected convolutional networks. 2017 IEEE conference on computer vision and pattern recognition (CVPR), 2261-2269 (2017) arXiv:1608.06993
22. D. Han, J. Kim, J. Kim, Deep pyramidal residual networks. 2017 IEEE conference on computer vision and pattern recognition (CVPR), 6307-6315 (2017) arXiv1610.02915
23. I.J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples. (2015) CoRR, abs/1412.6572. arXiv:1412.6572
24. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. Commun. ACM **60**, 84–90 (2012)
25. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**, 1929–1958 (2014)
26. C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals, Understanding deep learning requires rethinking generalization. (2017), arXiv:1611.03530
27. Shorten, C., Khoshgoftaar, T.M.: A survey on image data augmentation for deep learning. J. Big Data **6**, 1–48 (2019)
28. H. Touvron, A. Vedaldi, M. Douze, H. J'egou, Fixing the train-test resolution discrepancy: FixEfficientNet. (2020), arXiv:2003.08237
29. K. He, R.B. Girshick, P. Dollár, Rethinking ImageNet Pre-Training. 2019 IEEE/CVF international conference on computer vision (ICCV), 4917-4926 (2019) arXiv:1811.08883
30. P. Sun, R. Zhang, Y. Jiang, T. Kong, C. Xu, W. Zhan, M. Tomizuka, L. Li, Z. Yuan, C. Wang, P. Luo, Sparse R-CNN: end-to-end object detection with learnable proposals. 2021 IEEE/CVF conference on computer vision and pattern recognition (CVPR), 14449-14458 (2021) arXiv:2011.12450
31. S.H. Lee, S. Lee, B.C. Song, Vision transformer for small-size datasets. (2021), arXiv:2112.13492
32. H. Bao, L. Dong, F. Wei, BEiT: BERT pre-training of image transformers. (2021), arXiv:2106.08254
33. Fang, M., Chen, Z., Przystupa, K., Li, T., Majka, M., Kochan, O.: Examination of abnormal behavior detection based on improved YOLOv3. Electronics **10**, 197 (2021)
34. Song, W., Beshley, Przystupa, M., et al.: A software deep packet inspection system for network traffic analysis and anomaly detection. Sensors **20**(6), 1637 (2020)
35. Lu, X., Lu, X.: An efficient network for multi-scale and overlapped wildlife detection. SIViP (2022). https://doi.org/10.1007/s11760-022-02237-9
36. Borkar, M., Cevher, V., McClellan, J.H.: Low computation and low latency algorithms for distributed sensor network initialization. SIViP **1**, 133–148 (2007). https://doi.org/10.1007/s11760-007-0014-7

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.