Hindawi
Discrete Dynamics in Nature and Society
Volume 2020, Article ID 4706576, 11 pages
https://doi.org/10.1155/2020/4706576

Research Article

A Full Stage Data Augmentation Method in Deep Convolutional Neural Network for Natural Image Classification

Qinghe Zheng,[1] **Mingqiang Yang**,[1] **Xinyu Tian**,[2] **Nan Jiang**,[3] **and Deqiang Wang**[1]

[1]*School of Information Science and Engineering, Shandong University, Qingdao 266237, China*
[2]*College of Mechanical and Electrical Engineering, Shandong Management University, Jinan 250357, China*
[3]*School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430072, China*

Correspondence should be addressed to Mingqiang Yang; imageinstitute@outlook.com and Deqiang Wang; wdq_sdu@sdu.edu.cn

Received 2 November 2019; Accepted 16 December 2019; Published 11 January 2020

Guest Editor: Zheng Wang

Nowadays, deep learning has achieved remarkable results in many computer vision related tasks, among which the support of big data is essential. In this paper, we propose a full stage data augmentation framework to improve the accuracy of deep convolutional neural networks, which can also play the role of implicit model ensemble without introducing additional model training costs. Simultaneous data augmentation during training and testing stages can ensure network optimization and enhance its generalization ability. Augmentation in two stages needs to be consistent to ensure the accurate transfer of specific domain information. Furthermore, this framework is universal for any network architecture and data augmentation strategy and therefore can be applied to a variety of deep learning based tasks. Finally, experimental results about image classification on the coarse-grained dataset CIFAR-10 (93.41%) and fine-grained dataset CIFAR-100 (70.22%) demonstrate the effectiveness of the framework by comparing with state-of-the-art results.

1. Introduction

Computer vision is the first and most widely used field of deep learning technology. After the advent of AlexNet [1], deep convolutional neural networks (CNNs) have been quickly applied for various tasks in computer vision, including pedestrian detection [2], face recognition [3], image classification [4–6], semantic segmentation [7, 8], and target tracking [9, 10]. Due to the availability of big data and massive computing resources, overparameterized deep learning models have demonstrated their superior performance depending on the highly nonlinear fitting capabilities. So far, many kinds of deep learning models have been developed and improved, including different structures and connections [11]. The corresponding training methods are also constantly updated [12, 13].

However, deep learning still has many unintelligible properties and the theory behind it is not perfect. Typically, due to its difficulty of interpretation, deep learning models are difficult to be improved in a targeted manner. Researchers usually need to consider both optimization and generalization. Moreover, big data driven mode based deep CNNs still have the "overfitting" problem; that is, the neural network can perform well on the training set but cannot be effectively generalized on the unseen test data. On the other hand, a larger model tends to perform better [14], but it also requires people to make tradeoffs between accuracy and reasoning speed in practice. The noise in the natural image also affects the mining of implicit knowledge and the extraction of expressive features of the object. These challenges have hindered its successful application in some special scenarios, such as the medical diagnosis tasks [15], where there is lack of training data and automatic driving systems [16] that require high real-time performance.

At present, many methods have been developed to alleviate the "overfitting" problem of deep CNNs, and they can be summarized as follows:

(i) Regularization techniques for limiting network complexity, such as L2-regularization [17] and Hierarchical Guidance and Regularization (HGR) learning [18]

(ii) Data augmentation methods for expanding sample set, such as translation [19], horizontal flipping [20], and noise disturbance [21]

(iii) Model ensemble for reducing dependence on single network, for example, auxiliary classification nodes in GoogleLeNet [22], Dropout [23], and DropConnect [24]

(iv) Some special training tricks like well-designed initialization [25], early stopping [26], and learning rate decay [27]

In this paper, we propose the full stage (i.e., training and testing stages) data augmentation framework in deep learning for natural image classification. Data augmentation in the training process is used to ensure that the network can mine the structural information of samples and finally converge in the appropriate position, and the data augmentation in the test process can play the role of model ensemble to reduce the dependence on a single network. Augmentation in two stages needs to be consistent to ensure accurate transfer of domain information. It is worth noting that the framework is universal to any network architecture and data augmentation strategy and can therefore be applied to a variety of deep learning based tasks. We have done extensive experiments on fine-grained and coarse-grained image classification datasets, that is, CIFAR-10 and CIFAR-100 [28]. Compared with different algorithms, our framework shows significant improvement on deep CNN and achieves state-of-the-art results.

The remainder of the paper is organized as follows. Section 2 gives a brief review of the related work on data augmentation in deep learning. In Section 3, we introduce the proposed full stage data augmentation framework in detail. Experimental results and comparisons are presented in Section 4. Finally, we conclude our work and discuss future directions in Section 5.

2. Related Work

Data augmentation is an effective method to reduce the "overfitting" of deep CNN caused by limited training samples, which approximates the data probability space by manipulating input samples, such as horizontal flipping, random crop, scale transformation, and noise disturbance. In general, as long as the quantity, quality, and diversity of the data in the dataset are increased, the effectiveness of the model can be improved. Sample pairing [29] is a simple but surprisingly effective data augmentation technique for image classification task, which can create the new image from an original one by overlaying another image randomly picked from the training set. However, many special training tricks hinder its real application. Neural Augmentation [30] and Smart Augmentation methods [31] teach the neural network autonomous learning how to generate new samples by minimizing the error of that network. The appearance of Generative Adversarial Networks (GANs) provides a new research direction for data augmentation. Frid-Adar et al. [32] have illustrated that training with adversarial samples generated by GANs can improve the generalization ability of deep CNNs and help to overcome the defects of activation functions. But, in practice, GANs require considerable time for training and are difficult to converge. As for data augmentation in testing phase, Wang et al. [33] have used different underpinning network structures and augmented the image by 3D rotation, flipping, scaling, and adding random noise. Experiments showed that test-time augmentation can achieve higher segmentation accuracy and obtain uncertainty estimation of the segmentation results. There have been many data augmentation methods in deep learning community, but how to efficiently apply them is currently the most important research direction.

In addition, there are many regularization methods at the loss layer which can also be interpreted as an implicit data augmentation, such as Dropout [23], DropConnect [24], DisturbLabel [34], and SoftLabel [35]. Dropout and DropConnect can be interpreted as data augmentation methods by projecting the introduced noises back into the input space. DisturbLabel and SoftLabel add specially distributed noises to ground-truth category labels of randomly selected samples during the training process. The noises have been distributed in the implicit augmented samples. Although the above methods can improve the generalization ability of the model, the impact of additional noise on the decision boundary has not been analyzed rigorously.

In fact, approximating real and natural input spaces through data augmentation is intuitionistic. A more comprehensive input space allows the model to better converge on a global minimum or a better local minimum. However, the "overfitting" problem of deep CNNs still exists, which prompts us to rethink of the influence of data augmentations during training and testing process on the optimization and generalization of deep CNNs.

3. Full Stage Data Augmentation Framework

3.1. Problem Formulation. Given a deep CNN model $\mathbb{M}_0$: $f(\mathbf{x}; \boldsymbol{\theta}_0)$ trained on the training set D: $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$, $(\mathbf{x}, \mathbf{y})$ and $\boldsymbol{\theta}_0$: $\{\mathbf{W}_0^l, \mathbf{b}_0^l\}_{l=1}^{L}$ represent the inputs (i.e., the natural images and corresponding ground-truth labels) and initialized network parameters, respectively. Parameters are organized into four-dimensional tensors and two-dimensional matrices in the convolutional and fully connected layers, respectively. The network is optimized by mini-batch stochastic gradient descent (SGD) method based on back propagation.

In the forward propagation stage, the output of each layer is the input of the next; the output $\mathbf{h}_l$ of l-th layer in deep CNN for $l = 1, \ldots, L-1$ can be given by

$$\mathbf{h}_l = \sigma\left(\mathbf{W}_0^l \mathbf{h}_{l-1} + \mathbf{b}_0^l\right), \tag{1}$$

where $\mathbf{h}_0 = \mathbf{x}$ and $\sigma(\cdot)$ represents the element-wise nonlinear activation function, such as Leaky-ReLU [36], which is defined as

$$\sigma(x) = \begin{cases} x, & \text{if } x > 0, \\ \dfrac{x}{a}, & \text{if } x \le 0, \end{cases} \tag{2}$$

where a is a fixed hyperparameter in $(1, +\infty)$. Then the final output of deep CNN model can be obtained by

$$f(\mathbf{x}) = \text{softmax}\left(\mathbf{W}_0^L \mathbf{h}_{L-1} + \mathbf{b}_0^L\right), \tag{3}$$

where softmax $(\cdot)$ is defined as the logarithmic normalization function of finite term discrete probability distribution and can be calculated according to

$$\text{softmax}(f)_i = \frac{e^{f_i}}{\sum_{j=1}^{C} e^{f_j}}, \quad \text{for } i = 1, 2, \ldots, C, \tag{4}$$

where C is the number of neurons in the last layer, that is, the number of classification categories. Finally, the training loss of deep CNN can be given by

$$\mathscr{L}(\mathbf{x}_i, \mathbf{y}_i) = -\frac{1}{C} \sum_{j=1}^{C} \left[y_i^j \log f(\mathbf{x}_i)^j + \left(1 - y_i^j\right) \log\left(1 - f(\mathbf{x}_i)^j\right) \right] + \lambda \sum_{k=1}^{L} \left\| \mathbf{W}^k \right\|_F. \tag{5}$$

The first term is the negative log-likelihood loss and the second term is L2-regularization of all the weights. λ is the weight decay rate that controls the regularization intensity and $\|\cdot\|_F$ represents the Frobenius norm. By continuously optimizing the loss function and updating the network parameters, the model is trained for convergence and used for testing.

In the back propagation stage, our goal is to minimize $\mathscr{L}$ through updating parameters (weights $\mathbf{W}$ and biases $\mathbf{b}$) in deep CNN. Based on mini-batch SGD, parameters at t-th training iteration can be updated as

$$\mathbf{W}_t^l = \mathbf{W}_{t-1}^l - \alpha \cdot \frac{1}{M} \sum_{i=1}^{M} \frac{\partial \mathscr{L}(\mathbf{x}_i, \mathbf{y}_i)}{\partial \mathbf{W}_{t-1}^l},$$
$$\mathbf{b}_t^l = \mathbf{b}_{t-1}^l - \alpha \cdot \frac{1}{M} \sum_{i=1}^{M} \frac{\partial \mathscr{L}(\mathbf{x}_i, \mathbf{y}_i)}{\partial \mathbf{b}_{t-1}^l}, \tag{6}$$

where α and M represent the learning rate and batch size, respectively. Through continuous iteration (each of which includes M forward propagation steps and 1 back propagation step), a convergent model $\mathbb{M}_*$: $f(\mathbf{x}; \boldsymbol{\theta}_*)$ is obtained. In the test process, the convergent deep CNN model is used to output the category labels of test samples. Finally, the entire flow chart is drawn in Figure 1.

3.2. Data Augmentation during Training Process. From the perspective of image acquisition, an acquired image is only one of many possible observations of the potential anatomy that can be observed by different spatial transformations and noise disturbance. Direct inference of the acquired images may result in biased results affected by specific transformations and noise associated with equipment and environment. In order to obtain a more reliable and robust prediction, we propose a full stage data augmentation framework to decrease the "overfitting" problem in deep CNN.

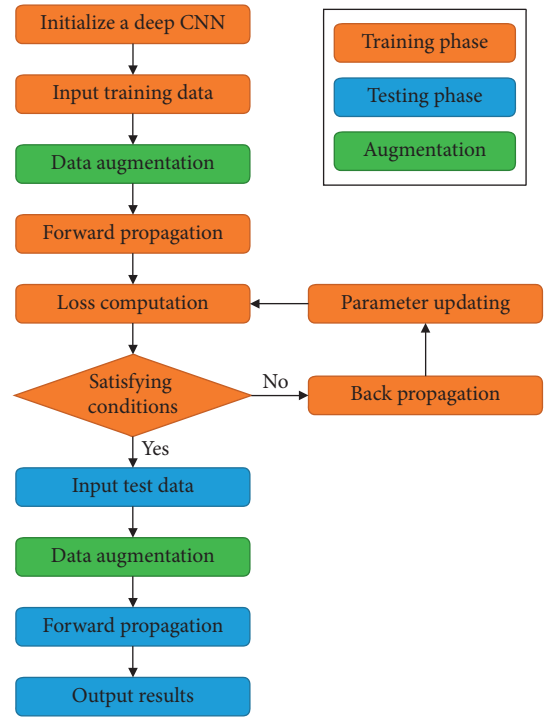

Figure 1: The overall flow of training and testing of deep CNNs.

At the first level, that is, training stage, the training samples D_t: $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{M}$ in a mini-batch set at t-th training iteration can be expanded to $\tilde{D}_t$: $\{(\tilde{\mathbf{x}}_i, \tilde{\mathbf{y}}_i)\}_{i=1}^{\tilde{M}}$ through various data augmentation techniques when they are fed into the deep CNN, such as translation [19], horizontal flipping [20], and noise disturbance [21]. All the augmentation parameters like translation step, rotation range, and noise intensity are set and retained. Furthermore, it is worth noting that all data augmentations are performed at the data input stage rather than at the beginning of the entire training process. In this way, the training data are expanded to $\tilde{M}/M$ times of the original data and the number of training iterations remains almost unchanged.

3.3. Data Augmentation during Testing Process. At the second level, that is, test stage, we use the same distributions of augmentation parameters for the convergent deep CNN. Each test image is augmented to $\tilde{M}/M$ images through the same data augmentations used in the training process. The consistency of data augmentation in the two stages is helpful to ensure the accurate transfer of domain information. The $\tilde{M}/M$ prediction results are combined to obtain the final prediction based on majority voting:

$$f(\mathbf{x}) = \frac{M}{\widetilde{M}} \sum_{i=1}^{\widetilde{M}/M} f(\widetilde{\mathbf{x}}_i). \tag{7}$$

Then the label corresponding to the location of the largest value in the one-dimensional vector $f(\mathbf{x})$ is the final prediction result. If there exists a balanced vote, the category with largest probability is chosen as the final prediction result. The whole framework is shown in Figure 2.

3.4. Interpretation as Model Ensemble. Researchers [37] have reported that the combination of deep CNNs trained on different noisy datasets is usually helpful. However, training each neural network separately is prohibitively expensive, since this requires exponentially many large sets containing noisy data. At test stage, data augmentation operation of each test sample ($\mathbf{x} \longrightarrow \widetilde{\mathbf{x}}$) can be viewed as $\widetilde{\mathbf{x}} = g(\mathbf{x})$ where $g(\cdot)$ represents corresponding data augmentation operation. Each different data augmentation strategy can be represented by a different $g(\cdot)$. Therefore, the final prediction based on majority voting can be rephrased as

$$f(\mathbf{x}) = \frac{M}{\widetilde{M}} \sum_{i=1}^{\widetilde{M}/M} f[g_i(\mathbf{x})] = \frac{M}{\widetilde{M}} \sum_{i=1}^{\widetilde{M}/M} \widetilde{f}_i(\mathbf{x}), \tag{8}$$

where $\widetilde{f}_i$ can be seen as a series of heterogeneous weak learners that focus on different aspects of training samples. Assuming that all the samples are independent and identically distributed (*i.i.d*), the data augmentation in the test stage can be interpreted as an implicit model ensemble through transforming $\widetilde{\mathbf{x}}$ back to $\mathbf{x}$. It can reduce the bias and variance of the convergent network, thus reducing the risk of "overfitting" problem on the training set while increasing the classification accuracy on the test set.

By reducing the reconstruction error between original sample and augmented samples, we can obtain the updated parameters of deep CNNs. We have observed the parameter distribution of a series of networks $[\widetilde{f}_1, \widetilde{f}_2, \ldots, \widetilde{f}_{\widetilde{M}/M}]$ in Figure 3. It can be seen that the parameter distribution of some networks is obviously different from that of other models. Therefore, these models can be viewed as focusing on different features of the image.

4. Experimental Results and Analysis

4.1. Experimental Setup

4.1.1. Experimental Datasets and Image Preprocessing. Two benchmarks CIFAR-10 and CIFAR-100 represent coarse-grained and fine-grained natural image classification tasks, respectively, which are used to evaluate the effectiveness of full stage data augmentation frameworks under different difficulties. CIFAR-10 and CIFAR-100 are labeled subsets of the 80 million tiny images dataset [28]. The CIFAR-10 dataset consists of 60,000 32×32 color images in 10 classes, with 6,000 images per class. There are 50,000 training images and 10,000 test images. CIFAR-100 is just like CIFAR-10, except it is a fine-grained version and has 100 classes containing 600 images each. There are 500 training images and 100 testing images in each class. Some examples of images in the two datasets are shown in Figure 4.

Input images of CIFAR-10 and CIFAR-100 datasets [28] are preprocessed in the following manner. Each original image is first color-normalized and then zero-padded to be 40×40 pixels. As for data augmentation at both training and testing stages, all samples are cropped to be 32×32 pixels and followed by a random horizontal flip with the 50% probability during both training and testing stages. The sample size is expanded ten times by considering model stability. Moreover, each image subtracts its own three-channel (R/G/B) mean value to speed up the convergence of deep CNN model.

4.1.2. Network Architectures. Two specially designed deep CNNs are constructed to complete the image classification, as shown in Figure 5. The network trained on CIFAR-100 uses a deeper and broader structure than network trained on CIFAR-10, because finer-grained data require a larger capacity for the model to characterize. Batch normalization layer [38] is added between each convolutional layer and the activation function. Fully connected layers that usually appear in traditional networks are replaced by global average pooling layer [39] to alleviate the "overfitting" problem, except for the last fully connected layer with softmax function used to output the category probability. All the weights in the network are set according to MSRA method [25].

4.1.3. Hyperparameters Setting. Network hyperparameters including initial learning rate, batch size, dropout rate, momentum, weight decay rate, and Leaky-ReLU hyperparameter a are set to 0.01, 512, 0.5, 0.9, 0.0005, and 5, respectively. Nine-tenths of the samples in a batch come from data augmentation. As training iterations, the learning rate is decreased in an exponential form with a decay rate of 0.9.

4.1.4. Experimental Platform. All the training and testing procedures of deep CNNs are carried out under the Caffe deep learning framework [40], based on the workstation consisting of an Intel Core i7-8700k CPU, a NVIDIA GeForce GTX 1080 GPU, 16 gigabytes of memory, and 1 terabyte of storage. The hardware platform and framework only affect the training efficiency rather than the actual classification performance of deep learning model.

4.2. Comparison of Classification Results. In this section, we report the experimental results and discuss possible reasons behind some phenomena. To prove the validity of proposed full stage data augmentation method, fivefold cross validation results are computed for final evaluation and comparison. Furthermore, the classification results of two datasets are presented separately in terms of the fineness degree of object categories.

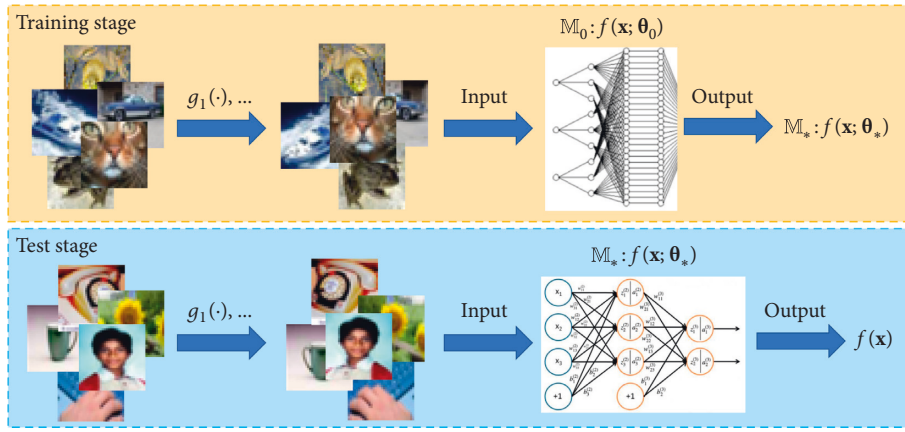

Figure 2: The whole full stage data augmentation framework.

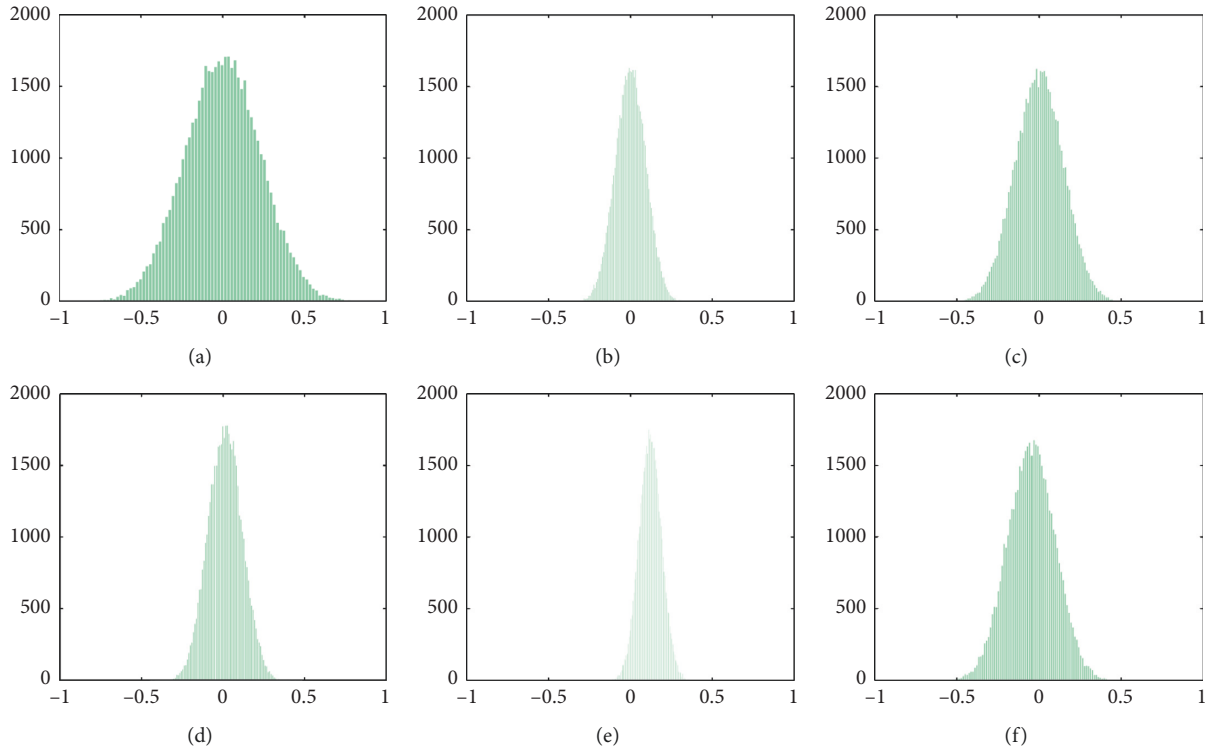

Figure 3: Parameter distribution of a series of deep CNNs by projecting the augmented images back into the input space. The horizontal axis represents the normalized network parameters.

4.2.1. Coarse-Grained Image Classification Results. We first report the baseline classification results before and after using full stage data augmentation method, as shown in Figure 6. The results show that the full stage data augmentation framework leads to a significant improvement of classification accuracy for deep CNN model. It can be clearly seen that the confusion matrix of original network is more confusing than that of the data-augmented network. In fact, the average classification accuracy of CIFAR-10 has increased from 85.7% to 93.4%. Furthermore, we also report the results of using data augmentation only during training or testing phase. Data augmentation in training phase is more effective than that in testing phase, with an accuracy increase of about 3%. However, the deep CNN also needs longer training time. In contrast, the additional reasoning costs associated with the data augmentation in the test phase can be neglected. In other words, full stage data augmentation framework improves the performance of traditional training data augmentation methods without introducing additional costs.

Then we observe the effectiveness of various data augmentation methods under the proposed full stage data augmentation framework, including translation, horizontal flip, rotation, scale transformation, and noise disturbance.

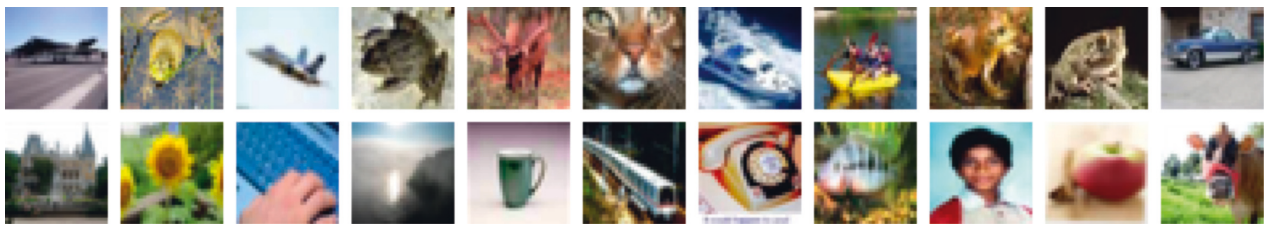

Figure 4: Some examples of images in CIFAR-10 (first row) and CIFAR-100 (second row).

CIFAR-10	CIFAR-100
32 × 32 × 3 input	32 × 32 × 3 input
3 × 3 conv, 64 3 × 3 conv, 64 Batch normalization	3 × 3 conv, 128 3 × 3 conv, 128 1 × 1 conv, 128 Batch normalization
2 × 2 max pool dropout (0.1)	2 × 2 max pool dropout (0.1)
3 × 3 conv, 128 3 × 3 conv, 128 Batch normalization	3 × 3 conv, 128 3 × 3 conv, 128 1 × 1 conv, 128 Batch normalization
2 × 2 average pool dropout (0.1)	2 × 2 average pool dropout (0.1)
3 × 3 conv, 128 3 × 3 conv, 128 Batch normalization	3 × 3 conv, 256 3 × 3 conv, 256 1 × 1 conv, 256 Batch normalization
Global average pool dropout (0.5)	Global average pool dropout (0.5)
Dense (10) Softmax	Dense (100) Softmax

Figure 5: The structure of two specially designed deep CNNs.

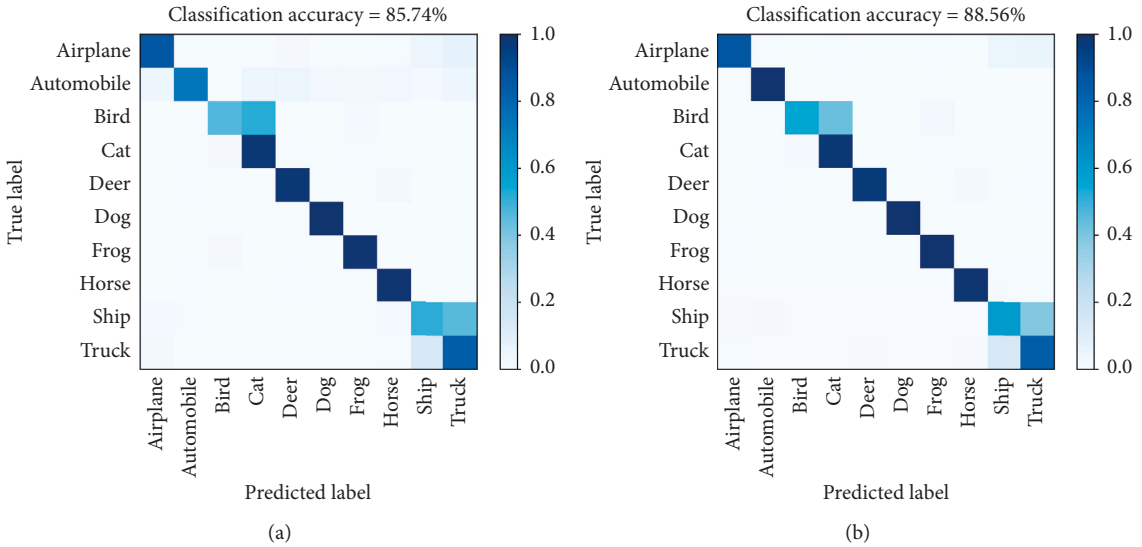

Figure 6: Continued.

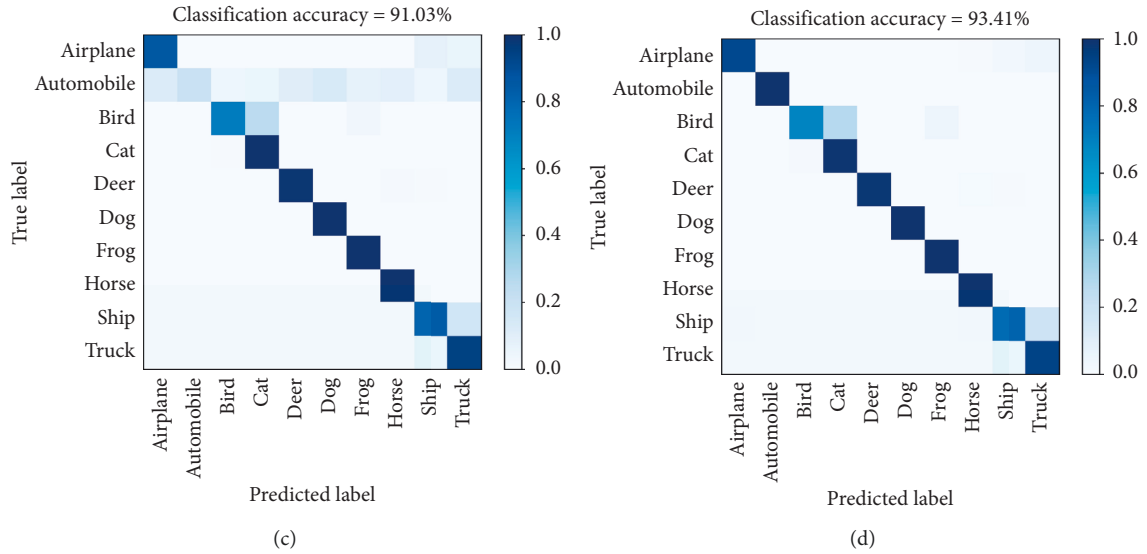

Figure 6: Classification results on CIFAR-10 before and after using full stage data augmentation method. (a)–(d) represent "no data augmentation," "augmentation in training stage," "augmentation in test stage," and "full stage data augmentation," respectively.

Translation and horizontal flip are based on the settings given above. The rotation range is from negative to positive five degrees, and the step size is 1 degree. Four Gaussian convolutional kernels with different fuzzy radii are used for the scale transformation, including 3×3, 5×5, and 7×7 pixels. The noise disturbance adds Gaussian white noises with different intensity to the original image, including 0.01, 0.05, 0.1, and 0.2. During the testing phase, the data augmentation strategy of test samples is consistent with the training samples. The experimental results are presented in Table 1. The results show that any data augmentation strategy under the full stage data augmentation framework can improve the classification performance of the deep CNN. As far as CIFAR-10 is concerned, translation and horizontal flip are the most effective means of data augmentation, while the improvement of classification accuracy caused by rotation and noise disturbance is limited. We think this may be related to the small size of the samples in CIFAR-10. The image itself is only 32×32 pixels and is quite blurred. Therefore, rotation and noise disturbance have a large impact on the image structure, resulting in limited help.

Finally, we compare state-of-the-art results brought by a series of algorithms to demonstrate the effectiveness of proposed full stage data augmentation framework, as shown in Table 2. These algorithms only adopt data augmentation strategies during the training phase. It can be seen that our proposed method has significantly improved the classification accuracy from 89.59% to 93.41%.

4.2.2. Fine-Grained Image Classification Results. Since its high similarity between different classes and the scarcity of samples in each class, fine-grained image classification is more challenging than coarse-grained classification task. Table 3 shows the experimental results on CIFAR-100 before and after using full stage data augmentation method. It can be seen that the average classification accuracy has been increased from 62% to 70%, which exceeds the improvement of CIFAR-10. On the other hand, the performance of single augmentation of training set and test set has also been improved, which increases the accuracy by 4.64% and 1.99%, respectively. In fine-grained image classification task, fewer samples in each classes caused by multiple classes make the data augmentation strategy play a greater role.

Then we also observe the effectiveness of various data augmentation methods on CIFAR-100, as given in Table 4. We find that the effect of data augmentation can only be achieved when the number of augmented samples becomes

Table 1: Classification accuracy of various data augmentation methods on CIAFR-10 under the proposed full stage data augmentation framework.

Methods	CIFAR-10 (%)
Translation	91.41
Horizontal flip	90.27
Rotation	88.78
Scale transformation	90.70
Noise disturbance	87.54

Table 2: Comparison with state-of-the-art algorithms on CIFAR-10.

Algorithms	CIFAR-10 (%)
Dropout [41]	84.40
Probout [42]	88.65
NIN + dropout [43]	89.59
Maxout + dropout [44]	88.32
Stochastic pooling [45]	84.86
Probabilistic weighted pooling [46]	88.71
Our method	93.41

Table 3: Experimental results on CIFAR-100 before and after using full stage data augmentation method.

Methods	CIFAR-100 (%)
No data augmentation	61.85
Augmentation in training stage	66.49
Augmentation in testing stage	63.84
Full stage augmentation	70.22

Table 4: Classification accuracy of various data augmentation methods on CIFAR-100 under the proposed full stage data augmentation framework.

Methods	CIFAR-100 (%)
Translation	63.73
Horizontal flip	64.11
Rotation	62.20
Scale transformation	64.83
Noise disturbance	60.47

larger than that of CIFAR-10. Moreover, fine-grained image classification is more sensitive to data augmentation strategy, and some methods may even have negative effects, such as the noise disturbance, which reduces the classification accuracy by 1.38%. This is related to the structure of dataset and the distribution condition of all samples. The distribution of samples in the dataset should be smooth; otherwise, it is easy to overlearn and cause the "overfitting" problem, which results in poor generalization on unseen test samples.

Table 5 gives the comparison results of CIFAR-100 with a series of state-of-the-art algorithms. It is worth noting that if dropout is employed improperly like in [45], the classification accuracy would decrease. Probabilistic weighted pooling [46] can also be regarded as model ensemble in the test stage, thus achieving good result (62.87%). Finally, classification accuracy has been increased from 64.32% to 69.22% by using full stage data augmentation framework.

4.3. Relationship between Data Augmentation and Network Generalization Ability. In practice, one of the obstacles to the mature application of data augmentation strategies in deep learning is that it is difficult for people to determine how many samples are efficient. In other words, the regularization intensity of data augmentation is usually uncertain. Although some scholars [47, 48] have suggested that the more samples the better, developers usually have to weigh the network performance and time cost in training and reasoning. In this part, we discuss the relationship between data augmentation and network generalization ability through extensive experiments.

We set up a series of data augmentation schemes of different sizes and observe the classification performance of the network in an attempt to mine and establish the relationship between the expanded sample size and the network generalization boundary. The experimental results of CIFAR-10 and CIFAR-100 are shown in Figure 7. The classification results of deep CNN with full stage data augmentation are always better than the baseline results on both datasets, regardless of the augmentation strength. In other words, the size of the dataset directly determines the quality of the deep learning models. On the other hand, the effect of various data augmentation methods clearly has a saturation interval. Once the augmentation strength exceeds this threshold, the performance of the network on the test set no longer grows and tends to be stable. In this case, we believe that the data itself or the network structure itself has become an "information bottleneck," which hinders the further improvement of classification accuracy. At this point, the direction of improvement should be considered from data sources with higher quality and more advanced network structures.

Table 5: Comparison with state-of-the-art algorithms on CIFAR-100.

Algorithms	CIFAR-100 (%)
Probout [42]	61.86
NIN + dropout [43]	64.32
Maxout + dropout [44]	61.43
Stochastic pooling [45]	57.49
Probabilistic weighted pooling [46]	62.87
Our method	70.22

Then we visualize the convolutional kernels in the first layer of deep CNN trained on CIFAR-10/100, as shown in Figure 8. All of them are ordered according to the value of their L1-norm. Visual spatial images can be combined by decoupled component-level convolutional kernels and mapped to different geometric spaces. These convolutional kernels reflect the organization information in the images extracted by the deep CNN, that is, the features of the object in the image. Generally speaking, the ordered convolutional kernels usually mean effective extraction of the organization information, while chaotic ones mean the "overfitting" of networks [49]. This is helpful for establishing the relationship between regularization intensity and network generalization ability and provides standards or principles to guide algorithm development or model structure improvement.

4.4. Impact on Network Optimization and Generalization. Data augmentation in training phase inevitably affects the network convergence, including the convergence speed and the final convergence position. We observe the decrease curve of the loss function of deep CNN on the training sets of CIFAR-10 and CIFAR-100 (see Figure 9) to analyze the impact of full stage data augmentation framework on network convergence. The loss of the model can reach the same level within the two epochs and eventually stabilize at 15 epochs, which means that the impact of data augmentation during the training phase on the convergence speed of the network can be ignored. On the other hand, the final loss value of the augmented model is slightly higher than that of the original model due to the regularization caused by the diversity of expanded samples. Actually, the generalization capability of deep CNNs, that is, the classification

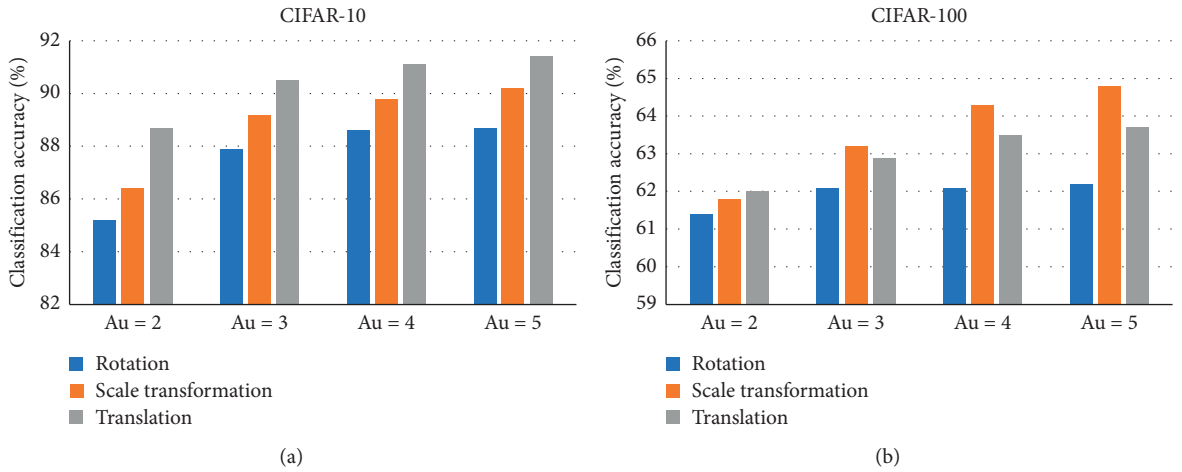

Figure 7: The relationship between the expanded sample size and network generalization ability, in which the number of images is expanded to Au times.

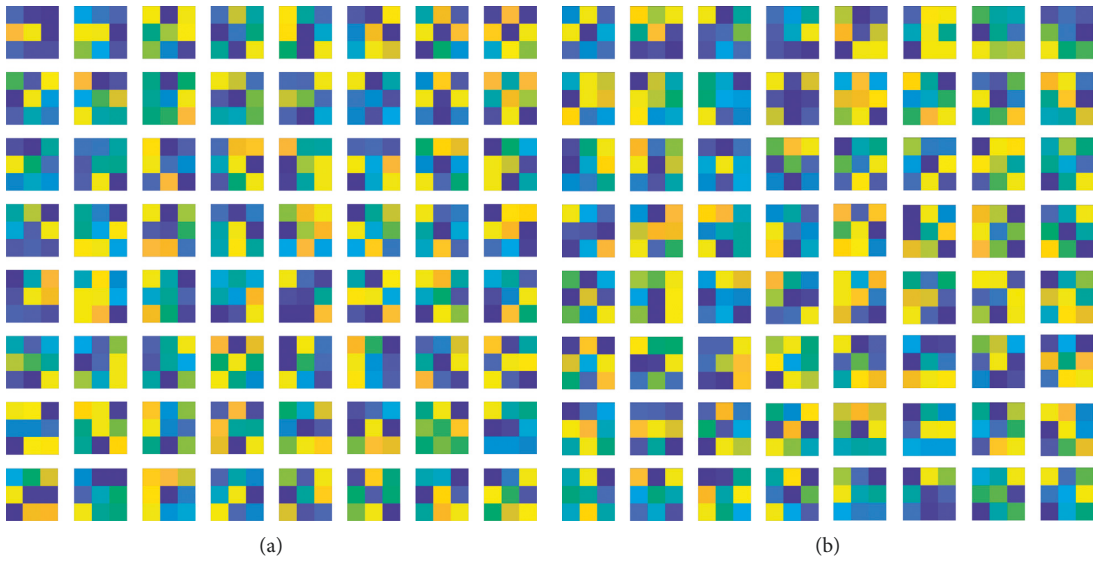

Figure 8: Visualization of convolutional kernels in the first layers of the deep CNNs trained on CIFAR-10 (a) and CIFAR-100 (b), respectively.

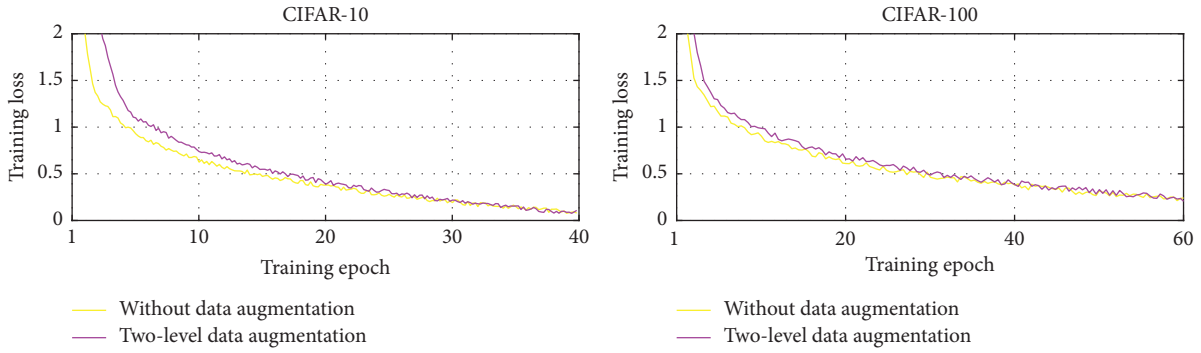

Figure 9: The optimization of the loss function of deep CNNs trained on CIFAR-10 and CIFAR-100, respectively.

performance on the test set rather than the training set, is our pursuit. Therefore, the optimization gap brought by data augmentation has no impact on the application of deep CNN in practice.

5. Conclusion

In this paper, we propose a full stage data augmentation framework to improve the accuracy of deep CNNs, which can also play the role of model ensemble without introducing additional model training costs. Simultaneous data augmentation during training and testing stages can ensure network convergence and enhance its generalization capability on unseen test samples. Furthermore, this framework is universal for any network architecture and data augmentation strategy and therefore can be applied to various deep learning based tasks. Finally, experiments about image classification on the coarse-grained dataset CIFAR-10 and fine-grained dataset CIFAR-100 demonstrate the effectiveness of the proposed framework by comparison with state-of-the-art algorithms. Through visualization of convolutional kernels, we have demonstrated that the ordered convolutional kernels usually mean effective extraction of the organization information, while chaotic ones mean the "overfitting" of networks. We have also analyzed the relationship between data augmentation and network generalization ability and observed the impact of the framework on the convergence of deep CNNs. The empirical results have shown that the data augmentation framework can improve the generalization ability of deep learning models, and it can have a negligible impact on the model's convergence.

As for future research directions, we plan to apply the proposed full stage data augmentation method to more complex CNN structures and some other machine learning related applications, such as liveness detection and gait and face recognition. We believe that it can help improve the performance of deep learning models in a series of tasks.

Data Availability

The experimental data of CIFAR-10 and CIFAR-100 used to support the findings of this study are included within the paper.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Key R&D Program of China (Grant 2018YFC0831503), the National Natural Science Foundation of China (Grant 61571275), and Fundamental Research Funds of Shandong University (Grant 2018JC040).

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 2, pp. 84–90, 2017.

[2] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, "Towards reaching human performance in pedestrian detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 973–986, 2018.

[3] J. Lu, G. Wang, and J. Zhou, "Simultaneous feature and dictionary learning for image set based face recognition," *IEEE Transactions on Image Processing*, vol. 26, no. 8, pp. 4042–4054, 2017.

[4] Q. Zheng, M. Yang, J. Yang, Q. Zhang, and X. Zhang, "Improvement of generalization ability of deep CNN via implicit regularization in two-stage training process," *IEEE Access*, vol. 6, pp. 15844–15869, 2018.

[5] Q. Zheng, M. Yang, Q. Zhang, and J. Yang, "A bilinear multi-scale convolutional neural network for fine-grained object classification," *IAENG International Journal of Computer Science*, vol. 45, no. 2, pp. 340–352, 2018.

[6] Q. Zheng, X. Tian, N. Jiang, and M. Yang, "Layer-wise learning based stochastic gradient descent method for the optimization of deep convolutional neural network," *Journal of Intelligent & Fuzzy Systems*, vol. 37, no. 4, pp. 5641–5654, 2019.

[7] F. S. Saleh, M. S. Aliakbarian, M. Salzmann, L. Petersson, J. M. Alvarez, and S. Gould, "Incorporating network built-in priors in weakly-supervised semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1382–1396, 2018.

[8] Y. Li, Y. Liu, G. Liu, D. Zhai, and M. Guo, "Weakly supervised semantic segmentation based on EM algorithm with localization clues," *Neurocomputing*, vol. 275, pp. 2574–2587, 2018.

[9] Q. Zhang, M. Liu, and S. Zhang, "Node topology effect on target tracking based on UWSNs using quantized measurements," *IEEE Transactions on Cybernetics*, vol. 45, no. 10, pp. 2323–2335, 2017.

[10] Q. Zheng, X. Tian, S. Liu et al., "Static hand gesture recognition based on Gaussian mixture model and partial differential equation," *IAENG International Journal of Computer Science*, vol. 45, no. 4, pp. 569–583, 2018.

[11] Q. Zheng, X. Tian, M. Yang, Y. Wu, and J. Su, "PAC-Bayesian framework based drop-path method for 2D discriminative convolutional network pruning," *Multidimensional Systems and Signal Processing*, 2019.

[12] J. Li, M. Yang, Y. Liu et al., "Dynamic hand gesture recognition using multi-direction 3D convolutional neural networks," *Engineering Letters*, vol. 27, no. 3, pp. 490–500, 2019.

[13] H. Zhuang, M. Yang, Z. Cui, and Q. Zheng, "A method for static hand gesture recognition based on non-negative matrix factorization and compressive sensing," *IAENG International Journal of Computer Science*, vol. 44, no. 1, pp. 52–59, 2017.

[14] D. Li, Y. Yang, Y. Song, and T. Hospedales, "Deeper, broader and artier domain generalization," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 5543–5551, Venice, Italy, 2017.

[15] Q. Zheng, M. Yang, Q. Zhang, X. Zhang, and J. Yang, "Understanding and boosting of deep convolutional neural network based on sample distribution," in *Proceedings of the IEEE Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, pp. 823–827, Chengdu, China, 2017.

[16] A. Gudigar, S. Chokkadi, and U. Raghavendra, "A review on automatic detection and recognition of traffic sign," *Multimedia Tools and Applications*, vol. 75, no. 1, pp. 333–364, 2016.

[17] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?," in *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, pp. 5574–5584, Long Beach, CA, USA, 2017.

[18] Z. Zhang, C. Xu, J. Yang, Y. Tai, and L. Chen, "Deep hierarchical guidance and regularization learning for end-to-end depth estimation," *Pattern Recognition*, vol. 83, pp. 430–442, 2018.

[19] Q. Zheng, X. Tian, M. Yang, and H. Wang, "Differential learning: a powerful tool for interactive content-based image retrieval," *Engineering Letters*, vol. 27, no. 1, pp. 202–215, 2019.

[20] T. Kobayashi, "Flip-invariant motion representation," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 5628–5637, Venice, Italy, 2017.

[21] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[22] C. Szegedy, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, Boston, MA, USA, 2015.

[23] N. Srivastava, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[24] L. Wan, M. Zeiler, S. Zhang, Y. LeCun, and R. Fergus, "Regularization of neural networks using DropConnect," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 1058–1066, Atlanta, GA, USA, 2013.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: surpassing human-level performance on ImageNet classification," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1026–1034, Santiago, Chile, 2015.

[26] Q. Zhang, A. Liu, and X. Tong, "Early stopping criterion for belief propagation polar decoder based on frozen bits," *Electronics Letters*, vol. 53, no. 24, pp. 1576–1578, 2017.

[27] G. A. Carpenter and W. D. Ross, "ART-EMAP: a neural network architecture for object recognition by evidence accumulation," *IEEE Transactions on Neural Networks*, vol. 6, no. 4, pp. 805–818, 1995.

[28] A. Krizhevsky, N. Vinod, and G. Hinton, "The CIFAR-10 dataset," 2014, http://www.cs.toronto.edu/kriz/cifar.html55.

[29] H. Inoue, "Data augmentation by pairing samples for images classification," in *Proceedings of the International Conference on Learning Representation (ICLR)*, pp. 313–322, Vancouver, BC, Canada, 2018.

[30] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 547–554, 2018.

[31] J. Lemley, S. Bazrafkan, and P. Corcoran, "Smart augmentation learning an optimal data augmentation strategy," *IEEE Access*, vol. 5, pp. 5858–5869, 2017.

[32] M. Frid-Adar, I. Diamant, E. Klang et al., "GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification," 2018, https://arxiv.org/abs/1712.04621.

[33] G. Wang, W. Li, S. Ourselin, and T. Vercauteren, *Automatic Brain Tumor Segmentation Using Convolutional Neural Networks with Test-Time Augmentation*, Springer, in *Proceedings of the International MICCAI Brainlesion Workshop*, pp. 61–72, Springer, Granada, Spain, September 2018.

[34] L. Xie, J. Wang, Z. Wei, M. Wang, and Q. Tian, "DisturbLabel: regularizing CNN on the loss layer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4753–4762, Las Vegas, NV, USA, 2016.

[35] H. Proenca, J. C. Neves, T. Marques, S. Barra, and J. C. Moreno, "Joint head pose/soft label estimation for human recognition in-the-wild," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 38, no. 12, pp. 2444–2456, 2016.

[36] X. Zhang, Y. Zou, and S. Wei, "Dilated convolution neural network with leaky-ReLU for environmental sound classification," in *Proceedings of the International Conference on Digital Signal Processing (DSP)*, pp. 1–5, London, UK, 2017.

[37] X. Tian, "A electric vehicle charging station optimization model based on fully electrified forecasting method," *Engineering Letters*, vol. 27, no. 4, pp. 731–743, 2019.

[38] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 448–456, Lille, France, 2015.

[39] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2921–2929, Las Vegas, NV, USA, 2016.

[40] Y. Jia, E. Shelhamer, J. Donahue et al., "Caffe," in *Proceedings of the ACM International Conference on Multimedia*, pp. 675–678, Orlando, FL, USA, 2014.

[41] G. E. Hinton, N. Srivastave, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaption of feature detectors," 2012, https://arxiv.org/abs/1207.0580.

[42] J. Springenberg and M. Riedmiller, "Improving deep neural networks with probabilistic maxout units," in *Proceedings of the International Conference on Learning Representations (ICLR)*, pp. 1–10, Banff, Canada, 2014.

[43] M. Lin, Q. Chen, and S. Yan, "Network in network," in *Proceedings of the International Conference on Learning Representations (ICLR)*, pp. 1–10, Banff, Canada, 2014.

[44] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout Networks," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 1319–1327, Atlanta, GA, USA, 2013.

[45] M. Zeiler and R. Fergus, "Stochastic pooling for regularization of deep convolutional neural networks," in *Proceedings of the International Conference on Learning Representations (ICLR)*, pp. 1–9, 2013, Scottsdale, AZ, USA.

[46] H. Wu and X. Gu, "Towards dropout training for convolutional neural networks," *Neural Networks*, vol. 71, pp. 1–10, 2015.

[47] G. Weisz, P. Budzianowski, P.-H. Su, and M. Gasic, "Sample efficient deep reinforcement learning for dialogue systems with large action spaces," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 11, pp. 2083–2097, 2018.

[48] S. Bianco, M. Buzzelli, D. Mazzini, and R. Schettini, "Deep learning for logo recognition," *Neurocomputing*, vol. 245, pp. 23–30, 2017.

[49] K. Knauf, D. Memmert, and U. Brefeld, "Spatio-temporal convolution kernels," *Machine Learning*, vol. 102, no. 2, pp. 247–273, 2016.