Two-Stage Selective Ensemble of CNN via Deep Tree Training for Medical Image Classification

Yun Yang, Yuanyuan Hu, Xingyi Zhang, *Senior Member, IEEE*, and Song Wang, *Senior Member, IEEE*

***Abstract*—Medical image classification is an important task in computer-aided diagnosis systems. Its performance is critically determined by the descriptiveness and discriminative power of features extracted from images. With rapid development of deep learning, deep convolutional neural networks (CNNs) have been widely used to learn the optimal high-level features from the raw pixels of images for a given classification task. However, due to the limited amount of labeled medical images with certain quality distortions, such techniques crucially suffer from the training difficulties, including overfitting, local optimums, and vanishing gradients. To solve these problems, in this article, we propose a two-stage selective ensemble of CNN branches via a novel training strategy called deep tree training (DTT). In our approach, DTT is adopted to jointly train a series of networks constructed from the hidden layers of CNN in a hierarchical manner, leading to the advantage that vanishing gradients can be mitigated by supplementing gradients for hidden layers of CNN, and intrinsically obtain the base classifiers on the middle-level features with minimum computation burden for an ensemble solution. Moreover, the CNN branches as base learners are combined into the optimal classifier via the proposed two-stage selective ensemble approach based on both accuracy and diversity criteria. Extensive experiments on CIFAR-10 benchmark and two specific medical image datasets illustrate that our approach achieves better performance in terms of accuracy, sensitivity, specificity, and F1 score measurement.**

***Index Terms*—Computer-aided diagnosis, convolutional neural networks (CNNs), deep learning, ensemble learning.**

I. Introduction

OVER the last decades, medical imaging, such as X-rays [1]; pathology imaging [2]; magnetic resonance imaging (MRI) [3], [4]; and computed tomography (CT) [5], has become a key technique for the early diagnosis and treatment of diseases. Having tremendous increase in medical images, it has become extremely hard to manually perform image-based medical examinations for physicians due to 1) the visual characteristics of medical images are not always easy to distinct, and some are visually different while others may be slightly different and 2) medical images are normally archived in large volume, high dimensionality, limited labeling information and the presence of quality distortion. On the other hand, such trend provides the potential for computer-aided diagnosis (CAD) systems that may revolutionize disease diagnosis and management by rapid examination of large number of images. CAD [6] can provide complementary objective advice to medical professionals for the assistance of diagnosis based on medical images. Previous, studies [7]–[9] have suggested that the incorporation of the CAD system into the diagnostic process can significantly improve the efficiency and effectiveness of image-based disease screening, and result in the decrease of interobserver variation [10] and the reduction of unnecessary false-positive biopsies [11] and thoracotomy [12].

Traditional image-based CAD approaches are commonly composed of feature extraction, feature selection and classification, in which feature extraction is the most critical step. Although many feature extraction techniques, such as filter-based features [13], [14]; scale-invariant feature transform (SIFT) [14], [15]; and local binary patterns (LBP) [16], [17] have been developed, they have to be manually set up, and require extensive tuning to accommodate specific tasks. Therefore, it is too expensive for medical image classification tasks.

Recent advances in deep learning offer an effective solution for image-based CAD [5], [18]. Deep learning [19] can extract high-level abstract features intrinsically from raw pixels of image data. As one of the typical deep learning algorithms, convolutional neural network (CNN) has shown the state-of-the-art performance in image classification tasks [20]. By adopting different convolutional kernels and network architectures, various CNNs, such as LENET [21], AlexNet [22], VGG net [23], Network in Network [24], GoogLeNet [25], ResNet [26], DenseNet [27], and CondenseNet [28] obtain the abstract representation of target image from different perspectives.

However, deep learning techniques still face general issues, such as weak labels and large images, for medical image classification tasks. Different from natural images, the biggest challenge of applying deep neural networks in medical images is that labeling massive medical images is extremely expensive and time consuming, hence the labeled medical images

Manuscript received 16 September 2020; revised 19 January 2021; accepted 19 February 2021. Date of publication 11 March 2021; date of current version 18 August 2022. This work was supported in part by the Natural Science Foundation of China (NSFC) under Grant 61663046 and Grant 61876166, and in part by the Yunnan Provincial Basic Research Program for Distinguished Young Scholar. This article was recommended by Associate Editor J. Han. *(Corresponding authors: Yun Yang; Xingyi Zhang.)*

Yun Yang and Yuanyuan Hu are with the National Pilot School of Software, Yunnan University, Kunming 650106, China (e-mail: yangyun@ynu.edu.cn; hyy.who@gmail.com).

Xingyi Zhang is with the Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, School of Computer Science and Technology, Anhui University, Hefei 230039, China (e-mail: xyzhanghust@gmail.com).

Song Wang is with the College of Engineering and Computing, University of South Carolina, Columbia, SC 29208 USA (e-mail: songwang@cec.sc.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCYB.2021.3061147.

Digital Object Identifier 10.1109/TCYB.2021.3061147

are always achieved in a limited amount for training the deep neural network, which usually causes the problem of overfitting. In fact, deep neural networks are susceptible for overfitting, in that learning model fits too well to the training set, and becomes difficult to generalize to new examples, such as testing set that are different from training set. Moreover, as the networks deepen, the training process becomes quite challenging due to the fundamental problem of vanishing and exploding gradients [29]–[31], which may make the network training to converge to bad local optimums or saddle points [32], and result in unsatisfied classification performance. On the other hand, conventional CNNs only focus on the deepest and the most abstract features, while few CNNs take the effectiveness of the middle layer features into account. In practice, medical images are normally obtained in the presence of various quality distortions and degradations, such as noise, blur, and compression effects, which cause more bias in the deeper layers [33], [34]. Therefore, middle-level features are also useful in the medical image classification tasks. Our previous studies [35], [36] show that the ensemble approach on the joint use of multiple feature representation obtained from the target dataset is more likely to achieve a global optimum solution for the learning tasks, and prevent the occurrence of overfitting.

To overcome these problems, in this article, we propose a novel method for medical image classification via combining the deep CNN and ensemble approach. In summary, our contributions are highlighted as follows.

1) Inspired by the nature process of that plant roots nourish the branches, a novel training strategy called deep tree training (DTT) is proposed to supplement gradients from deeper layers to shallower layers via splitting multiple branches from the hidden layers in order to solve the problem of vanishing gradients.
2) To incorporate the middle-level features into the medical image classification task, a novel ensemble approach is proposed to obtain the multiple classifiers on the middle-level features during DTT training, which effectively prevents the problems of overfitting and local optimums.
3) We propose a two-stage selection scheme to automatically select the optimal ensemble members from branch classifiers based on accuracy and diversity criteria, and further adopt weighting strategy to consolidate them in order to achieve the optimal ensemble classifier.

The remainder of this article is organized as follows. We review the related work in Section II. Section III describes our approach with technical details of the proposed DTT and the two-stage selective ensemble approach. Section IV presents our experiments on medical image classification tasks. Finally, we draw conclusions in the last section.

II. Related Work

CNN has been widely used for medical image classification tasks, such as mammography mass lesion classification [37], automatically detection of invasive ductal carcinoma [38], Alzheimer's disease classification [25], and so on. Many works [24], [39], [40] have shown that the image-based CAD based on CNN generally outperforms the conventional machine learning methods. But they always require more training tips to prevent the problems of vanishing gradients, local optimums and overfitting in the medical image classification tasks. To tackle these problems, four strategies have been proposed in previous works. The first one is a strategy of greedy layerwise training [41] for solving the problem of local optimums appeared in the gradient-based optimization process. The second strategy solves the problem of vanishing gradients by using unsaturated neurons [42]–[44] to replace traditional saturated neurons, and the third strategy is to supplement gradients for hidden layers [25]–[27], [45]. The fourth strategy aims to solve the problem of overfitting via a variety of regularization techniques called "sparse constraint" [46], "dropout" [47], "batch normalization" [48], "shakeout" [49], and "maxout" [50]. Despite that these methods can solve the problems to some extent, there is not a universal solution to all the problems, and few of them takes into account the importance of middle-level features for classification tasks.

Similar to our approach, several CNN models, including GoogLeNet [25], deeply supervised nets (DSN) [45], ResNet [26], and DenseNet [27] have been developed for overcoming the problem of vanishing gradients from different aspects. GoogLeNet [25] branches two additional softmax classifiers from hidden layers in the process of training, but those branch classifiers are discarded during testing. Its branch classifiers have one pooling layer, one convolutional layer and two full connection layers, whereas our branch classifiers only have one full connection layer. In the training process of DSN, a branch classifier is also associated with each hidden layer, where the backpropagation does not only from the final layer but simultaneously from the hidden layers, and then learning output is still determined by the final output layer. In contrast, our approach takes the ensemble of multiple branch classifiers as the final classifier. While ResNet adopts an identity mapping to establish a shortcut connection that skips one or more layers, where bypasses the input information to the output and protecting the integrity of the information, the entire network only needs to learn the part of the input and output differences called residual, simplifying the learning objectives and difficulty. Actually, ResNet is a fusion of multiple shallow networks. It does not fundamentally solve the vanishing gradient problem, but avoids such problem due to that it is composed of multiple shallow networks. DenseNet is to establish the dense connection between each layer and its previous layers, where the feature mapping of the current layer is passed to all subsequent layers, so that each layer gets the collective knowledge of all the previous layers via cascading manner. Therefore, in the process of backpropagation, the front layer can also obtain the gradient supplement of all subsequent layers, thereby alleviating the problem of vanishing gradients.

Our approach has several major differences with above methods: 1) the gradient supplement strategy of our approach is proposed in an iterative manner, and involves all the hidden layers from deep to shallow, which can provide much more supplement to the shallow layers; 2) our approach adopts multiple middle-level features to conduct classification task via ensemble learning, while GoogLeNet, DSN, ResNet only use the most abstract features for classification task; 3) in DenseNet, using middle-level features can be

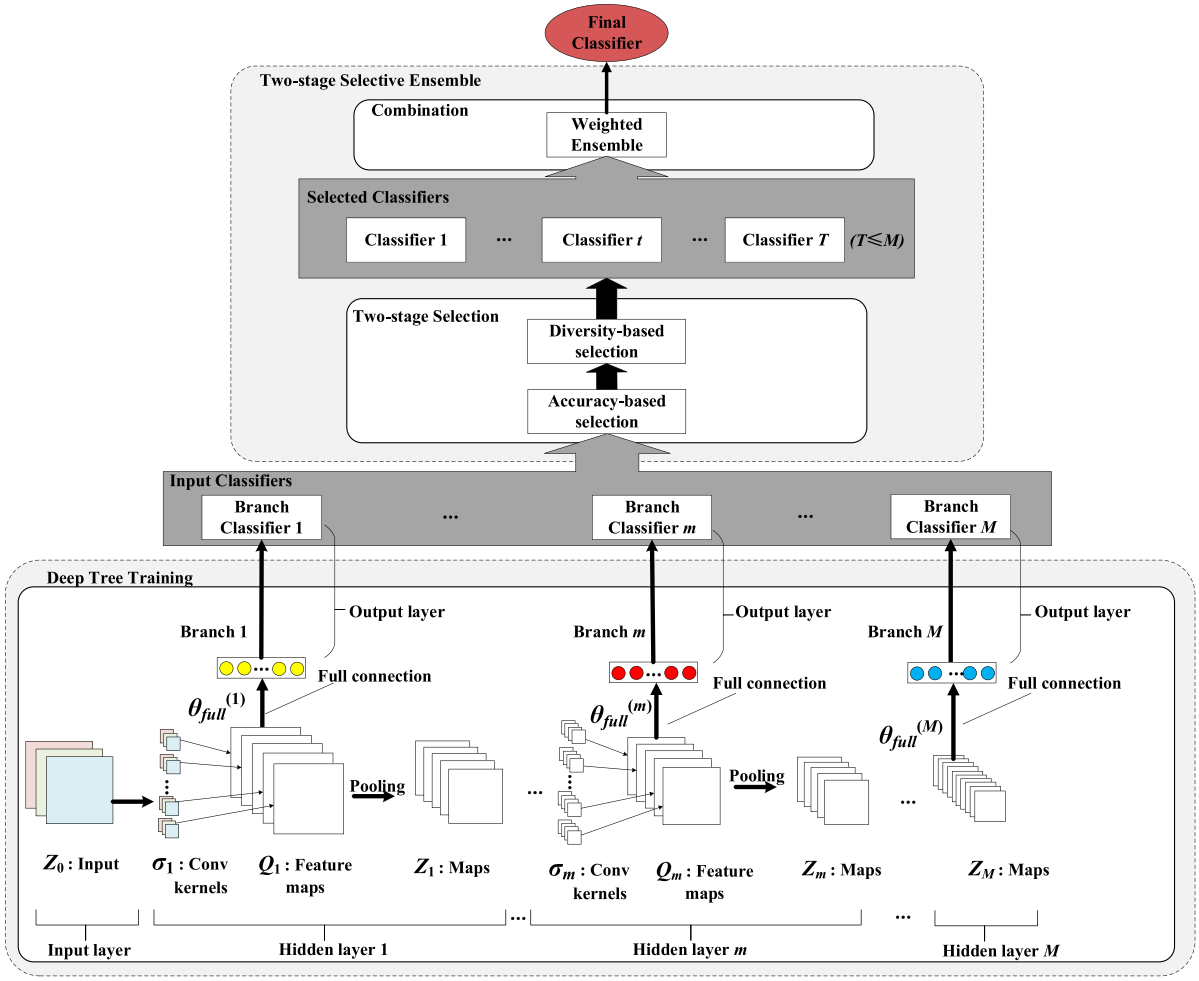


Fig. 1. Overview of our approach. This figure shows the two modules of our approach with the traditional CNN network architecture. In the DTT module, we extend a branch classifier from each hidden layer. Each branch classifier is partially overlapping but has a different structure of the diverse learner and they can be obtained in one training session. In the two-stage selective ensemble module, we ultimately select the base learners from multiple branch classifiers for ensemble solution via "accuracy" and "DF diversity" criteria.

seen as integration at the feature level, while our approach is to integrate multiple branch classifiers obtained on middle-level features via ensemble learning, and to take full advantage of each hidden layer feature; and 4) our approach is easy to implement, and compatible with most of existing CNNs for building better classifiers. This has been tested on network in network (NIN) and GoogLeNet, and the results reported in Section IV clearly demonstrate that the classification performance of both NIN and GoogLeNet have been significantly improved by incorporating our approach.

Ensemble learning [51], [52] refers to building multiple different and effective base learners as ensemble members, and then integrating them to make decisions, so as to achieve better performance than any single base learner. Due to the advantages of ensemble learning, three major ensemble learning strategies of bagging [53], boosting [54], and stacking [55] have been proposed for different learning tasks, including supervised learning [56]–[59], unsupervised learning [35], [36], [60], [61], semisupervised learning [62], and achieved promising results. Furthermore, previous studies [63] and [64] have proved the efficacy of ensemble learning in medical image classification tasks. For example, Fraz *et al.* [65] designed a supervised ensemble method for segmenting of blood vessels in retinal photographs, Ochs *et al.* [66] utilized boosting to build a set of weak classifiers and then integrated them to classify pulmonary structures and obtained improved results, and Termenon and Graña [67] proposed a sequential ensemble method to classify Alzheimer's disease based on MRI and obtained better results.

The key to the success of ensemble learning is to build a set of accurate and diverse base classifiers. However, it is not straightforward to determine a tradeoff state between average accuracy and diversity of ensemble members for the optimal ensemble performance. Moreover, there is not a well-accepted formal definition for measuring the diversity. On the other hand, the implementation of deep learning coupled with ensemble learning approaches is extremely time consuming, and as such the direct application of their combination to classify massive images is almost infeasible in practical situation.

III. Method

In this section, an overview of our approach is given at the beginning, and then followed by the technical details of the

proposed DTT scheme and the two-stage selective ensemble approach.

A. Overview

As illustrated in Fig. 1, our approach is comprised of two modules, the DTT and the two-stage selective ensemble, they are described as follows.

1) In the DTT module, multiple branches of CNN are obtained via iteratively attaching an output layer to the hidden layers of deep CNN, and these branches are trained to not only optimally supplement the gradients from the deeper layers to the shallower layers but also create a set of classifiers on the middle-level features, which just happens to produce the inputs of ensemble module without additional cost. The technical details of DTT are presented in Section III-B.
2) In the two-stage selective ensemble module, a two-stage selection scheme is adopted to select the optimal ensemble members from the branch classifiers due to the fact that some of branch classifiers have the degradation effect to the ensemble solution. It selects the classifiers with high accuracy as the candidates in the first stage, and further selects the ones with high diversity as ensemble members from the candidates in the second stage. Finally, a weighted ensemble method is used to combine the ensemble members into final classifier by considering different contributions of the ensemble members to the final decision. The technical details of the two-stage selective ensemble approach are presented in Section III-C.

B. Deep Tree Training

The major difficulty in training deep neural network using the gradient-based optimization is the vanishing gradients problem, which is caused by the chain rule computation of gradient on small value during the backpropagation process. In order to alleviate this problem, we design a novel training strategy called DTT, which intends to inject gradients for hidden layers by simulating the process of that a tree transports nutrition from roots to branches. As illustrated in Fig. 1, CNN with DTT is split into M branches by adding the output layers into the hidden layers, then the gradients of layer m is strengthened by adding up the gradients from all the branches from m to M. Therefore, the shallower layers will get more gradient supplement.

Here, we denote the input training dataset as $S = \{(X_i, Y_i), i = 1, 2, \ldots, N\}$, in which $X_i \in R^n$ denotes the input image data and $Y_i \in \{1, \ldots, K\}$ is the corresponding label for sample X_i. In this illustration, σ_m and υ_m refer to the convolution kernels and bias term for hidden layer m, respectively, and combining them gives $w_m = (\sigma_m, \upsilon_m)$. We can denote the convolution and activation processes in layer $m \in \{1, \ldots, M\}$ as

$$Q_m = f(\sigma_m * Z_{m-1} + \upsilon_m), \text{ and } (Z_0 \equiv X_i) \tag{1}$$

where M is the total number of hidden layers, $w_m$is the weights for hidden layer m, while $*$ corresponds to the convolution operation, Z_{m-1} refers to the feature maps compressed by a pooling function at the hidden layer $m-1$, and Q_m denotes the feature maps after convolutions and activation processes. Some recent studies [22] suggested that rectified linear (ReLu) function $f(\mu)$=max(0, μ) improves both learning speed and classification performance in CNN application, and hence we choose such function as the neuron activation function. Subsequently, the compressed feature maps Z_m is obtained from feature maps Q_m at the hidden layer m via the max pooling function, $Z_m = g(Q_m)$.

In our approach, an additional output layer is attached to each hidden layer of deep CNN, where the feature maps are fed into a softmax classifier [68] via the full connection layer. In each branch network, we utilize $V^{(m)}$ and $w^{(m)}_{\text{full}}$ to represent the input vector and the weights matrix of full connection, respectively. $V^{(m)}$ refers to the flatten features of feature map Q_m obtained from the hidden layer m. After full connection layer, the input vector $O^{(m)}$ of the output layer in branch m can be obtained as

$$O^{(m)} = w^{(m)}_{\text{full}} V^{(m)} + b^{(m)}_{\text{full}} \tag{2}$$

where $b^{(m)}_{\text{full}}$ is a bias term defined in the output function of full connection layer. Finally, a softmax function $h^{(m)}(\cdot)$ is attached at the end of branch m to establish the branch classifier m

$$h^{(m)}(y = k|X_i) = \frac{\exp\left(O^{(m)}_k\right)}{\sum_{j=1}^{K} \exp\left(O^{(m)}_j\right)}, \text{ and } \forall k \in \{1, \ldots, K\} \tag{3}$$

where $O_j^{(m)}$ refers to the jth element of input vector $O^{(m)}$, and y denotes the possible category of sample X_i. Giving an example of classification task on Alzheimer's disease, $h^{(m)}(y$='AD'$|X_i)$ calculates the probability of that sample X_i is diagnosed as AD by branch classifier m. The output of $h^{(m)}(\cdot)$ is a probability vector of instance X_i classified as different categories. The class label is then determined by the category with the highest probability.

The cross-entropy loss [19] is usually used to measure the dissimilarity between predicted label and true class label in the classification task, where the smaller value of cross-entropy loss corresponds to higher similarities between the predicted label and true label, and hence the better classification performance is achieved. By taking all the weight matrix and bias term of the full connection layer in branch classifier m as $\theta^{(m)}_{\text{full}} = \left(w^{(m)}_{\text{full}}, b^{(m)}_{\text{full}}\right)$, such loss function is defined as

$$\text{Loss}^{(m)}\left(W^{(m)}, \theta^{(m)}_{\text{full}}\right) = -\sum_{i=1}^{n}\sum_{k=1}^{K} Y_i \log\left(h^{(m)}(y = k|X_i)\right). \tag{4}$$

Here, $W^{(m)} = (w_1, \ldots, w_m)$ is established by concatenating the convolution weights of hidden layer *1* to *m*. *n* is the number of instances, and $1 \le n \le N$.

In general, mini-batch stochastic gradient descent is used to train the neural networks with deep architecture since it is a better choice to use the average value as the unbiased

estimation of loss. Therefore, the objective function of branch classifier m can be expressed as

$$J^{(m)}\left(W^{(m)}, \theta_{\text{full}}^{(m)}\right) = \min_{W^{(m)}, \theta_{\text{full}}^{(m)}} \left(\frac{1}{n}\text{Loss}^{(m)}\left(W^{(m)}, \theta_{\text{full}}^{(m)}\right)\right). \quad (5)$$

By summing up the objectives of all branch classifiers, the objective function of our approach is finally obtained as

$$J(W, \theta_{\text{full}}) = \min_{W, \theta_{\text{full}}} \left(\sum_{m=1}^{M} \beta^{(m)} J^{(m)}\left(W^{(m)}, \theta_{\text{full}}^{(m)}\right)\right) \quad (6)$$

in which $W = (W^{(1)}, \ldots, W^{(M)})$ and $\theta_{\text{full}} = (\theta_{\text{full}}^{(1)}, \ldots, \theta_{\text{full}}^{(M)})$, and $\beta^{(m)}$ denotes the weights of branch classifier m. In our approach, we simply use the average weighting scheme $\beta^{(m)}$=M^{-1} in order to avoid additional computational cost, which denotes that each branch classifier effects training process equally. Essentially, DTT can be formulated as optimizing the objective function defined in (6) and such optimization process intrinsically generates a set of branch classifiers on different middle-level features, which is regarded as the input candidates of the two-stage selective ensemble module.

Backpropagation is a generalization of the delta rule to multilayered feedforward networks, made possible by using the chain rule to iteratively compute gradients. Then, the network parameters are updated with the computed gradients. In our approach, the parameters' gradient $\nabla\theta_{\text{full}}^{(m)} = (\nabla w_{\text{full}}^{(m)}, \nabla b_{\text{full}}^{(m)})$ of additional output layers can be calculated as

$$\begin{aligned}\nabla\ \theta_{\text{full}}^{(m)} &= \frac{\partial J(W, \theta_{\text{full}})}{\partial \theta_{\text{full}}^{(m)}} \\ &= \beta^{(m)} \times \frac{\partial J^{(m)}\left(W^{(m)}, \theta_{\text{full}}^{(m)}\right)}{\partial h^{(m)}} \times \frac{\partial h^{(m)}}{\partial O^{(m)}} \times \frac{\partial O^{(m)}}{\partial \theta_{\text{full}}^{(m)}}. \quad (7)\end{aligned}$$

Here, the weights $w_{\text{full}}^{(m)}$ can be update at the ith iteration as $w_{\text{full}}^{(m),i} = w_{\text{full}}^{(m),i-1} - \lambda\nabla w_{\text{full}}^{(m),i}$, where λ is the learning rate. Similarly, the update of $b_{\text{full}}^{(m)}$ at the ith iteration can be expressed as $b_{\text{full}}^{(m),i} = b_{\text{full}}^{(m),i-1} - \lambda\nabla b_{\text{full}}^{(m),i}$.

For the CNN without DTT, the entire network only has one output layer and the gradient ∇w_m of hidden layer m can be obtained as

$$\begin{aligned}\nabla\ w_m &= \frac{\partial J(W, \theta_{\text{full}})}{\partial w_m} \\ &= \frac{\partial J\left(W^{(M)}, \theta_{\text{full}}^{(M)}\right)}{\partial h^{(M)}} \times \frac{\partial h^{(M)}}{\partial O^{(M)}} \times \frac{\partial O^{(M)}}{\partial V^{(M)}} \\ &\quad \times \frac{\partial V^{(M)}}{\partial Z_M} \times \frac{\partial Z_M}{\partial Q_M} \cdots \times \frac{\partial Q_m}{\partial w_m}. \quad (8)\end{aligned}$$

Then, we can update the parameters w_m of the hidden layer m at the ith iteration as $w_m^i = w_m^{i-1} - \lambda\nabla w_m^i$. However, the computation of ∇w_m is a continued product from the output layer to the hidden layer m, with small value of derivative obtained at each layer. As the network goes deeper, the gradient of shallow layers may get vanished. As a result, the training process becomes unfunctional. This is the vanishing gradients problem commonly appeared in the training process of deep neural networks.

In contrast, by training CNN with DTT, M (M>1) branches are obtained as trimmed version of CNN with less numbers of hidden layers. In the backpropagation process, the gradients fed back by each branch are combined at the meeting point in a summing manner [19], therefore, in each branch, the parameters' gradients can be calculated by using gradient descent method shown in (8), then the gradient ∇w_m^* of hidden layer m is obtained by summing the gradients obtained in all the branches from m to M

$$\begin{aligned}\nabla\ w_m^* &= \frac{\partial J(W, \theta_{\text{full}})}{\partial w_m} \\ &= \sum_{i=0}^{M-m} \beta^{(m+i)} \times \frac{\partial J^{(m+i)}\left(W^{(m+i)}, \theta_{\text{full}}^{(m+i)}\right)}{\partial w_m} \\ &= \sum_{i=0}^{M-m} \beta^{(m+i)} \times \frac{\partial J^{(m+i)}\left(W^{(m+i)}, \theta_{\text{full}}^{(m+i)}\right)}{\partial h^{(m+i)}} \times \frac{\partial h^{(m+i)}}{\partial O^{(m+i)}} \\ &\quad \times \frac{\partial O^{(m+i)}}{\partial V^{(m+i)}} \times \frac{\partial V^{(m+i)}}{\partial Z_{m+i}} \times \frac{\partial Z_{m+i}}{\partial Q_{m+i}} \cdots \times \frac{\partial Q_m}{\partial w_m}. \quad (9)\end{aligned}$$

In the above equation, $([\partial J^{(m+i)}(W^{(m+i)}, \theta_{\text{full}}^{(m+i)})]/[\partial w_m])$ indicates the gradient of hidden layer m obtained in branch m+i, and $\beta^{(m+i)}$ refers to the weights of branch classifier m+i, and again we use average weighting scheme in our approach. It can be observed that the gradient ∇w_m^* is greater than the ∇w_m, which effectively alleviate the vanishing gradients problem to some extent.

In fact, the essence of DTT is to obtain the average of gradients from different branch classifiers with a common objective, and use these supplementary gradients to update the parameters of network for minimizing objective function. The average gradients can significantly reduce the impact of contingency, and lead to a robust training process for deep neural network, which has been demonstrated by its experimentally observed advantage in effectiveness and efficiency.

C. Two-Stage Selective Ensemble

Previous, study [52] suggests that it is better to partially combine the accurate and diverse base learners instead of all for an efficient ensemble solution. Therefore, to achieve a promising classification performance, we further propose a two-stage selective ensemble approach. It selects a part of branch classifiers obtained from middle-level features as ensemble members at two stages by considering their accuracy and diversity. Then, the selected ensemble members are optimally consolidated into the final classifier via a weighted ensemble strategy.

1) Accuracy-Based Selection: The fundamental criterion of selecting the ensemble members is the accuracy of the base learners. It has been proven that better performance of ensemble could be achieved by rejecting weak learners. In the first stage of selection, we first measure the accuracy of branch classifiers on validation set, and then access the validation accuracy of branch classifiers by sorting them in descending order. Only the top R classifiers are selected as candidates for the second stage of selection.

Algorithm 1: Diversity-Based Selection

Input: validation dataset $S_{\text{val}} = \{X_i\}_{i=1}^{N}$, the number of selected classifiers T and candidate classifier set $D = \{d_i\}_{i=1}^{M}$

Output: selected classifier set $D^* = \{d_i^*\}_{i=1}^{T}$

```
Compute the error rate of each classifier d_i ∈ D on S_val
d ← the classifier achieved the lowest error rate on S_val
D* ← {d}
D ← D/D*
for t = 1 to T − 1 do
    for each classifier d_j ∈ D do
        Compute Div(d_j, D*) as defined in (11)
    R_t ← sorted classifiers d_j ∈ D in ascending order by Div(d_j, D*)
    d ← top 1 classifier in R_t
    D* ← D* ∪ {d}
    D ← D/{d}
    return D*
```

2) Diversity-Based Selection: It is well known that to gain from combination, the ensemble members must be different, and otherwise there would be no performance improvement if they are identical. Thus, diversity has been recognized as an important characteristic in ensemble learning. Although there is no formal definition of diversity in the field of machine learning, various diversity measurements have been proposed from different perspectives, including pairwise and nonpairwise method. Shipp and Kuncheva [69] found that double fault (DF) [70] is an effective measurement of diversity via studying the correlation between diversity measurement and ensemble method. It indeed represents the statistical relationship between a pair of base classifiers. As shown in Table I, the DF-based diversity between classifiers d_i and d_j can be calculated by the joint distribution of such pair of classifiers

$$\text{Div}(d_i, d_j) = N^{00} \Big/ \left(N^{11} + N^{10} + N^{01} + N^{00}\right) \tag{10}$$

where N^{00} refers to the number of samples that are incorrectly classified by both classifier d_i and d_j on validation dataset. Analogously, N^{10} is the number of samples which are correctly classified by d_i but incorrectly classified by d_j on validation dataset. In fact, DF-based diversity indicates the intersection of misclassification areas between a pair of base classifiers: 1) the smaller the value of DF and 2) the higher the ensemble accuracy. The diversity between a based classifier d_i and a set of classifiers D^* can be further obtained as

$$\text{Div}(d_i, D^*) = \frac{1}{|D^*|} \sum_{d_j \in D^*} \text{Div}(d_i, d_j). \tag{11}$$

As illustrated in Algorithm 1, it begins to form the initial ensemble member set of having a single classifier with the best accuracy on the validation dataset S_{val}, and then iteratively selects the base classifier d from the candidate set D obtained at the first selection stage to add into the ensemble member set D^* with maximum diversity measured in (11). As suggested by Martínez-Muñoz *et al.* [71], we select 20%–50% of candidate set to form the ensemble member set in our diversity-based selection stage.

TABLE I
JOINT DISTRIBUTION OF A PAIR OF BASE CLASSIFIERS

	$d_j(X_k) = Y_k$	$d_j(X_k) \neq Y_k$
$d_i(X_k) = Y_k$	N^{11}	N^{10}
$d_i(X_k) \neq Y_k$	N^{01}	N^{00}

3) Combination of Selected Classifiers: By considering different contributions of the selected classifiers to the ensemble solution, a weighted ensemble model is used in our approach, and it can be formulated as

$$EN(x) = \sum_{t=1}^{T} \alpha_t d_t(x) \tag{12}$$

where T denotes the number of selected classifiers in ensemble set D^*, and $\alpha_t (\alpha_t \geq 0 \sum_{t=1}^{T} \alpha_t = 1)$ is the weight for classifier d_t in D^*, while $d_t(x)$ is the predicted result of the t-th classifier in D^*, and $EN(x)$ represents the output of the weighted ensemble model. The objective of our weighted ensemble approach can be formulated as minimizing the following loss function:

$$\begin{aligned}
\text{Loss}(EN) &= \sum_{x} \left\| \sum_{t=1}^{T} \alpha_t d_t(x) - f(x) \right\|^2 \\
&= \int \left(\sum_{t=1}^{T} \alpha_t d_t(x) - f(x) \right)^2 p(x) dx \\
&= \int \left(\sum_{i=1}^{T} \alpha_i d_i(x) - f(x) \right) \\
&\quad \times \left(\sum_{j=1}^{T} \alpha_j d_j(x) - f(x) \right) p(x) dx \\
&= \sum_{i=1}^{T} \sum_{j=1}^{T} \alpha_i \alpha_j CQ_{ij}
\end{aligned} \tag{13}$$

where $CQ_{ij} = \int (d_i(x) - f(x))\big(d_j(x) - f(x)\big) p(x) dx$, and the instance x of the training set is sampled according to the distribution $p(x)$. It is obvious that the optimal weight can be solved by [52]

$$a = \text{argmin}_a \sum_{i=1}^{T} \sum_{j=1}^{T} \alpha_i \alpha_j CQ_{ij}. \tag{14}$$

By adopting the Lagrange multiplier method [52], the weight of classifier d_t can be finally obtained as

$$\alpha_t = \sum_{j=1}^{T} CQ_{jt}^{-1} \Big/ \sum_{i=1}^{T} \sum_{j=1}^{T} CQ_{ij}^{-1}. \tag{15}$$

In the above equation, $\sum_{j=1}^{T} CQ_{jt}^{-1}$ corresponds to the quality of classifier d_t, and it has to be normalized by $\sum_{i=1}^{T} \sum_{j=1}^{T} CQ_{ij}^{-1}$.

IV. EXPERIMENTS AND EVALUATION

To validate our proposed approach for the image classification task, we conduct our experiments and evaluation

TABLE II
NIN WITH DTT CONFIGURATIONS

Layer	Type	Num Kernels	Kernel size	Stride	Activation
0	Input	3	img_size	-	-
1	Conv	64	7*7	2	ReLu
2	Max pooling	-	3*3	2	-
A1	Softmax	-	-	-	-
3	Conv	192	5*5	1	ReLu
A2	Softmax	-	-	-	-
4	Conv	160	1*1	1	ReLu
5	Conv	96	1*1	1	ReLu
A3	Softmax	-	-	-	-
6	Max pooling	-	3*3	2	-
7	Dropout	-	-	-	-
8	Conv	192	5*5	1	ReLu
A4	Softmax				
9	Conv	192	1*1	1	ReLu
10	Conv	192	1*1	1	ReLu
A5	Softmax	-	-	-	-
11	Max pooling	-	3*3	2	-
12	Dropout	-	-	-	-
13	Conv	192	3*3	1	ReLu
A6	Softmax	-	-	-	-
14	Conv	192	1*1	1	ReLu
A7	Softmax				
15	Conv	3	1*1	1	ReLu
16	Average pooling	-	8*8	1	-
A8	Softmax	-	-	-	-

TABLE III
GOOGLENET WITH DTT CONFIGURATIONS

Layer	Type	Num Kernels	Kernel size	Stride	Activation
0	Input	3	img_size	-	-
1	Conv	64	7*7	2	ReLu
2	Max pooling	-	3*3	2	-
3	LocalRespNorm	-	-	-	-
4	Conv	64	1*1	1	ReLu
5	Conv	192	3*3	1	ReLu
6	LocalRespNorm	-	-	-	-
7	Max pooling	-	3*3	2	-
A1	Softmax	-	-	-	-
8	Inception	-	-	-	ReLu
A2	Softmax	-	-	-	-
9	Inception	-	-	-	ReLu
A3	Softmax	-	-	-	-
10	Max pooling	-	3*3	2	-
11	Inception	-	-	-	ReLu
A4	Softmax	-	-	-	-
12	Inception	-	-	-	ReLu
A5	Softmax	-	-	-	-
13	Inception	-	-	-	ReLu
A6	Softmax	-	-	-	-
14	Inception	-	-	-	ReLu
A7	Softmax	-	-	-	-
15	Inception	-	-	-	ReLu
A8	Softmax	-	-	-	-
16	Max pooling	-	3*3	2	-
17	Inception	-	-	-	ReLu
18	Inception	-	-	-	ReLu
19	Average pooling	-	7*7	1	-
20	Dropout	-	-	-	-
A9	Softmax	-	-	-	-

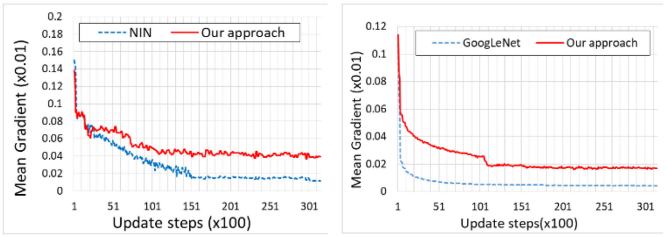
on three image datasets, including: 1) CIFAR-10 benchmarking dataset [72]; 2) breast histopathology images (BHI) dataset [73]; and 3) Chest X-rays dataset [74]. The first part of our experiments is carried out to evaluate the general performance of our approach on the CIFAR-10 dataset due to the fact of that such dataset is well accepted for benchmarking, and most of well-established deep learning methods have been validated on CIFAR-10. In the second part, we evaluate our approach in real-world medical applications by processing micrograph of invasive ductal carcinoma (IDC) and chest X-rays images for pneumonia. In our experiments, we intend to verify the own ability of the CNN-based models to overcome overfitting under the condition of limited training set, because data augmentation is extremely capable of relieving overfitting, which is not conducive to our experiments and analysis. Therefore, the testing approaches are performed on the original datasets without data augmentation. While pretrained models on ImageNet were not adopted due to that we intend to evaluate the performance of compared approaches with insufficient training data.

A. Experimental Setup

1) Testing Environment Setup: Our experiments are implemented by python 3.6 and Keras (high-level API on the top of TensorFlow) with GPU of NIVIDIA 1080ti. For comprehensive evaluation on medical image datasets, we adopt four performance measures in classification tasks, including: 1) classification accuracy; 2) sensitivity; 3) specificity; and 4) F1 score. Apart from the classification accuracy and F1 score which are standard validation metrics, sensitivity, and specificity are essential in medical diagnosis analysis: they indicate the misdiagnosis rate and missed diagnosis rate, respectively. Furthermore, we study the binary classification performance of the compared methods on medical image datasets by observing area under curve (AUC) on receiver operating characteristic (ROC) curves. In our experiments, we compare our proposed approach with a variety of recently well-established CNN models, including AlexNet [22], VGG net [23], NIN [24], GoogLeNet [25], ResNet(depth=110) [26], and DenseNet (k=12, depth=40) [27], and their implementations are obtained from GitHub.

2) DTT Setup: With the flexibility of our approach, we apply DTT to both NIN and GoogLeNet, which have shown good performance for a variety of image classification tasks [63]. As shown in Tables II and III, two approaches are comprised of eight branches (A1–A8) and nine branches (A1–A9), respectively, completed with softmax at the end, and a branch classifier is attached to each hidden layer. In our NIN-based approach, multilinear perceptrons are adopted within the receipt field instead of traditional linear filters, and the global average pooling layer is used to get rid of the full connection layer at the end thereby reducing parameters and complexity. In our GoogLeNet-based approach, the "Inception v1" modules are adopted, which is a subnetwork comprises of parallel convolutional filters whose outputs are concatenated. The repetition of the Inception v1 modules captures the optimal sparse representation of the image while simultaneously reducing dimensionality.

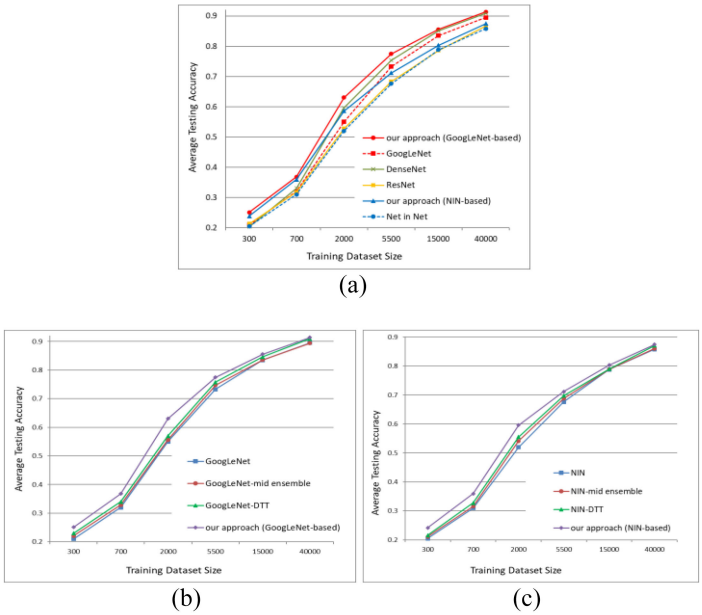
3) Hyperparameters Setup: The experiment datasets are divided into training, validation, and testing dataset, where the



(a) (b)

Fig. 2. Mean absolute value of gradients of weights and biases during the training on CIFAR-10 dataset. (a) NIN and our approach (NIN-based) comparison. (b) GoogLeNet and our approach (GoogLeNet-based) comparison.

validation dataset is specially used to determine hyperparameters of testing approaches. In our experiments, we train our approach with learning rate of 1e-3 at 100 epochs, and adopt the mini-batch stochastic gradient descent method to update the weights of our approach with batch size of 128. In addition, we apply the dropout method in our approaches, where the hyperparameter called keep probability is selected as 0.5. In the ensemble learning module, the number of ensemble member candidates R is set to 6 in the accuracy-based selection, and the number of ensemble members T is selected as 4 during the diversity-based selection.



(a) (b) (c)

Fig. 3. Classification on CIFAR-10 using training sets with different size. The average testing accuracies of these models are obtained by 5 runs of the experiment. (a) Performance of our approaches and other competing methods. (b) Performances of our NIN-based approach and its submodels. (c) Performances of our GoogLeNet-based approach and its submodels.

B. Experiment With CIFAR-10 Dataset

To evaluate the effectiveness of our approach from different perspectives, and compare with state-of-the-art techniques, we first conduct experiment on the CIFAR-10 dataset. The CIFAR-10 dataset consists of 32×32 color images from ten different classes, including 50 000 training and 10 000 testing images. Further, we divide original training set into training set (80%) and validation set (20%). In this part of our experiments, we conduct three experiments to test our approach from three perspectives, the capability of alleviating vanishing gradients, preventing overfitting and improving classification performance stuck at bad local optima.

First, we trained our approaches and their prototypes with average weighted ensemble and their average values of gradients at shallowest layer during training process are recorded, and presented in Fig. 2(a) and (b). These figures show that our approaches backpropagate stronger gradient-based feedback than their prototypes due to the gradients supplement of branch classifiers.

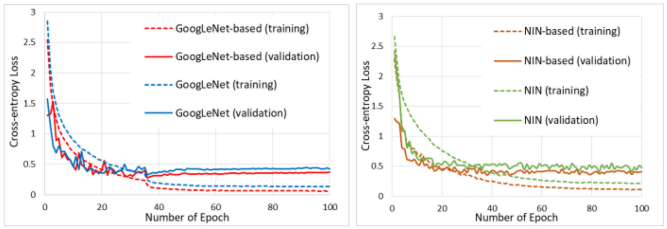
Second, we follow the experiment protocol in [49] to investigate the capability of preventing overfitting for compared approaches, where the performance of the approaches trained on different sizes of training set are validated with their average testing accuracy. As shown in Fig. 3(a), our approaches are generally superior to others, and has a larger lead gap in testing accuracy when the amount of training data is smaller. Because the deep learning model trained on the limited training set tends to be overfitting and results a poor testing accuracy. Such experiment results demonstrate that our approaches have better resistance to overfitting than their prototypes and state-of-the-art approaches. Actually, our approach incorporates gradient injection via DTT and middle-level features via ensemble learning manner to prevent overfitting. In order to explore the contribution of each component to the improvement on classification accuracy, we disassemble our approach into two submodels: 1) multiple middle level feature, respectively, obtained from hidden layers of NIN and GoogLeNet is directly fed into several attached softmax classifier as base learner, and then a final classifier is achieved by an ensemble of these base learners. They are called as NIN-mid ensemble and GoogLeNet-mid ensemble in Fig. 3(b) and (c) and 2) similar to DSN by using softmax [45], a branch classifier is attached to each hidden layers of NIN and GoogLeNet for gradient injection, and finally the deepest classifier result is taken, they are called as NIN-DTT and GoogLeNet-DTT in Fig. 3(b) and (c). By incorporating both of gradient injection and ensemble of middle level features, our approaches (NIN-based and GoogLeNet-based) are finally conducted. It can be seen from Fig. 3(b) and (c) that our approaches have better performances than others in the present of limited training data. A close observation indicates that the DTT-based methods (NIN-DTT and GoogLeNet-DTT) outperform their corresponding middle level feature ensemble-based methods (NIN-mid ensemble and GoogLeNet-mid ensemble). Such results can reveal that the effectiveness of gradient injection via DTT is more obvious, and the sole integration of the middle layer of the prototype model has no significant improvement on classification performance. Because these intermediate layers are far from the final output layer, the objective function is weakly supervised and cannot be directly used for classification tasks. As we know, the primary cause of overfitting in deep neural network approach is the insufficient training data. These experiment results show that our approach has significant capability of preventing overfitting caused by the limited training data.

TABLE IV
CIFAR-10 CLASSIFICATION ACCURACY (%)

Methods	ACC
AlexNet[22]	84.52±0.12
Net in Net[24]	87.93±0.09
VGG[23]	85.35±0.08
GoogLeNet[25]	92.79±0.05
ResNet[26]	89.34±0.05
DenseNet[27]	92.67±0.04
NIN- mid ensemble (Conventional)	86.94±0.06
NIN- mid ensemble (Our)	87.99±0.07
NIN-DTT	89.11±0.08
Our approach (NIN-based)	89.74±0.03
GoogLeNet-mid ensemble(Conventional)	92.81±0.04
GoogLeNet-mid ensemble(Our)	92.88±0.03
GoogLeNet-DTT	93.05±0.06
Our approach (GoogLeNet-based)	**93.62±0.01**

TABLE V
EXTRA 5 TESTING DATASETS INFORMATION

Testing set	Patches	Positive rate (%)
P1	1594	28.04
P2	842	51.31
P3	917	59.11
P4	1039	59.77
P5	638	69.12



(a) (b)

Fig. 4. Cross-entropy loss of approaches from training and validation on CIFAR-10, are calculated by (4). (a) Loss of our NIN-based approach and its prototype model. (b) Loss of our GoogLeNet-based approach and its prototype model.

TABLE VI
PERFORMANCES COMPARISON (%) OF THE COMPETING APPROACHES ON IDC DATASET

Methods	ACC	SEN	SPE	F1
AlexNet[22]	84.55±0.71	84.64±1.26	87.01±0.98	78.78±0.79
Net in Net[24]	85.16±0.54	84.01±0.96	87.27±0.85	79.08±0.58
VGG[23]	83.77±0.69	85.26±1.02	85.92±0.81	78.42±0.65
GoogLeNet[25]	85.23±0.49	83.95±0.78	88.95±0.75	78.74±0.52
ResNet[26]	87.54±0.51	86.87±0.69	92.32±0.73	85.37±0.50
DenseNet[27]	87.62±0.45	86.86±0.72	90.88±0.66	85.22±0.56
Our approach (NIN-based)	87.03±0.30	86.38±0.36	90.79±0.43	85.60±0.30
Our approach (GoogLeNet-based)	**88.80±0.25**	**88.56±0.33**	**93.16±0.36**	**86.33±0.22**

Third, we follow the experiment protocol in [75] to investigate the performance of approaches on local optima. We trained our approaches along with their prototypes on 40 000 images as training set, and validated them on 10 000 images as validation set, the cross-entropy loss of approaches from training and validation are calculated by (4), respectively. As shown in Fig. 4, our approaches and their prototypes are able to converge at certain local optima, however, our approaches achieve better local optima, which are illustrated by the smaller losses on training and validation, than their prototype models with faster convergence speed. Such results significantly demonstrate our approaches are able to prevent worse local optima to some extent.

The last part of this experiment is conducted for 5 runs. Table IV shows the mean and standard deviation of the classification accuracy on the predefined testing set of 10 000 samples. As illustrated, the both of NIN and GoogLeNet-based submodels equipped with our two-stage selective ensemble learning approach achieve high classification accuracy than the one with conventional ensemble learning setup of that all branch classifiers are combined via average weighting scheme, such result significantly demonstrate the effectiveness of our proposed two-stage selective ensemble learning scheme. Meanwhile the submodels, in association with either two stage selective ensemble learning or DTT module, are superior to their prototypes (NIN and GoogLeNet), and our approaches achieve the best performance by incorporating both proposed modules, where the higher average results illustrate that our approach overwhelmingly outperform its baselines and other competing methods, while the smaller value of standard deviation achieved by our approaches demonstrate their stable classification performance due to the fact that our approaches are insensitive to model initialization, and prevent a poor result stuck at local optima.

C. Experiment With Medical Image Datasets

In this part of experiments, we are going to systematically investigate the feasibility of our approach for medical image classification tasks from different perspectives.

1) Breast Histopathology Images Dataset: IDC [76] is the most common type of breast cancer: almost 80% of all breast cancers are IDCs. However, due to the lack of discriminative features, it is difficult to determine IDC as a special histological type like lobular or tubular cancer. In this experiment, we evaluate the performance of our approaches in comparison with six CNNs on the BHI dataset [73]. This dataset comprises 274 histopathology slide images of IDC tissue regions, which are digitized by a whole-slide scanner, from 274 patients. Each slide is split into nonoverlapping patches of 50*50 pixels via grid sampling, and the patches with fatty tissue or slide background are discarded, and then the networks take input image size of 50*50. The class labels of patches have been manually annotated by a pathologist.

This experiment is conducted on two subsets sampled from entire BHI dataset with ready processed image patches. The first subset consists of 269 patients' slides with 272 494 patches, including 76 303 positive samples and 196 191 negative samples as shown in Fig. 5. To evaluate the general classification performance of the compared approaches, we divided the patch images of such subset into training set (60%), validation set (20%), and testing set (20%). On the other hand, we simulate the practical diagnosis by applying our approaches on the second subset consisting of

TABLE VII
DIAGNOSIS PERFORMANCES (%) OF THE COMPETING APPROACHES ON THE 5 PATIENTS' SLIDE OF IDC DATASET

Testing set	Measures	*AlexNet[22]*	*Net in Net[24]*	*VGG[23]*	*GoogLeNet[25]*	*ResNet[26]*	*DenseNet[27]*	*Our approach*	
								NIN-based	***GoogLeNet-based***
P1	ACC	88.21±1.03	89.90±0.99	90.90±0.95	91.84±0.87	92.62±0.89	92.78±0.91	92.35±0.67	**94.10±0.75**
	SEN	93.51±1.38	90.45±0.94	90.05±0.87	90.05±0.83	90.69±0.78	**95.53±0.85**	92.96±0.56	87.47±0.44
	SPE	86.14±0.97	90.41±0.87	90.85±0.89	92.71±0.78	**99.22±0.85**	91.72±0.77	92.72±0.68	96.69±0.75
	F1	81.64±0.90	82.25±0.76	84.88±0.81	85.90±0.74	87.87±0.79	88.13±0.83	87.32±0.57	**89.27±0.52**
P2	ACC	86.82±1.26	84.56±0.56	85.99±0.93	86.48±0.51	89.33±0.92	89.42±0.78	89.31±0.36	**91.09±0.33**
	SEN	87.50±0.98	84.26±0.92	86.11±0.86	93.06±0.72	94.02±0.67	**94.44±0.81**	93.98±0.53	92.59±0.45
	SPE	83.10±0.77	82.88±0.69	82.85±0.71	79.02±0.60	**96.83±0.69**	86.46±0.75	85.61±0.32	89.51±0.49
	F1	87.20±0.88	84.85±0.87	86.31±0.87	87.24±0.79	90.60±0.81	90.08±0.71	90.02±0.61	**91.43±0.59**
P3	ACC	78.95±1.24	88.84±1.04	80.95±1.17	81.66±0.98	89.51±0.92	90.51±0.88	**91.60±0.81**	88.66±0.84
	SEN	65.50±0.96	87.39±0.84	66.87±0.82	70.45±0.76	74.72±0.85	85.46±0.73	**89.88±0.63**	85.42±0.67
	SPE	95.40±1.07	95.73±0.92	94.87±0.90	94.40±0.87	94.54±0.91	**97.87±0.89**	97.20±0.79	93.33±0.71
	F1	78.63±0.98	86.25±0.93	78.72±0.87	88.11±0.81	90.04±0.88	**91.41±0.76**	89.26±0.72	89.90±0.69
P4	ACC	83.16±1.01	91.63±0.97	93.07±0.94	92.30±0.88	94.45±0.93	**94.61±0.79**	93.92±0.74	94.03±0.71
	SEN	88.57±0.99	88.08±0.91	88.73±0.85	90.18±0.82	90.68±0.87	**93.88±0.81**	90.98±0.67	90.50±0.58
	SPE	72.73±0.81	96.89±0.78	97.52±0.74	97.85±0.73	99.32±0.77	95.69±0.73	98.30±0.59	**99.28±0.49**
	F1	86.49±0.68	92.63±0.51	92.87±0.57	91.22±0.49	89.30±0.63	95.42±0.55	**95.74±0.24**	94.77±0.36
P5	ACC	86.52±1.13	90.28±1.09	91.63±1.02	91.48±0.97	95.03±0.89	96.67±0.84	94.98±0.93	**97.02±0.83**
	SEN	83.45±1.27	91.46±0.77	92.52±0.73	92.74±0.71	82.77±0.79	94.56±0.72	95.69±0.62	**97.28±0.58**
	SPE	91.40±0.97	88.68±0.91	92.89±0.84	91.88±0.85	**99.49±0.82**	94.92±0.89	95.43±0.71	96.45±0.79
	F1	89.54±0.61	92.14±0.57	92.55±0.59	93.46±0.48	90.46±0.53	96.08±0.51	96.13±0.35	**97.83±0.31**

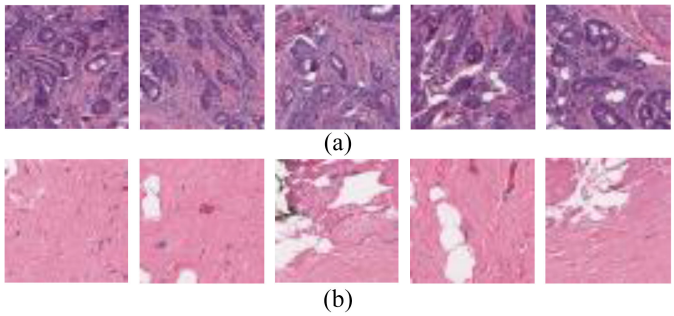


Fig. 5. IDC Dataset. (a) Positive samples (IDC). (b) Negative samples (non-IDC).

five patients' slide images with different rates of positive samples shown in Table V.

These experiments are conducted for five runs, and the results are presented in form of mean and standard deviation.

As shown in Table VI, among the compared approaches, our GoogLeNet-based approach achieves the best performance in terms of accuracy, sensitivity, specificity, and F1 score. From the perspectives of different measures, it can be seen that the six CNNs without ensemble module produce higher standard deviations, indicating that the performances of such CNNs are unstabilized due to the fact that they are more sensitive to model initialization than our approaches. From these results, we can see that our approach provides a promising technique for binary classification tasks on medical image dataset.

To evaluate our approach in practical diagnosis, Table VII lists all the testing results achieved on the five sets of slides obtained from five patients with different positive rate shown in Table V, where these compared models are trained by the same setup in the previous part of experiment. It is observed that, our GoogLeNet-based approach win on P1, P2, and P5 sets with two or three out of four validation criteria, while our NIN-based approach and DenseNet achieved the best results with two validation criteria on P3 set, respectively, and our approaches and DenseNet win on P4 with two validation criteria, respectively. Such results clearly demonstrate the effectiveness of our approach in the real-world medical application. In addition, according to the classification accuracy, we selected the top three competing methods, including: 1) ResNet; 2) DenseNet; and 3) our approach (GoogLeNet-based), and further evaluate their performance by plotting their ROC curves with AUC indicator. As shown in Fig. 6, our approach is closest to the upper left corner, and the AUC area is larger than the other two methods. This further illustrates that our approach is superior over other methods in medical applications.

2) Chest X-Rays Dataset: According to World Health Organization (WHO), pneumonia [77] is the single leading cause of childhood mortality, especially in developing countries, such as Southeast Asia and Africa. In fact, bacterial and viral pathogens are two major causes of pneumonia, but require very different manners of treatment and management. Therefore, how to accurately diagnose different types of pneumonia is very important to treat such disease. As a major technique, chest X-rays imaging is commonly used to diagnose pneumonia patients. However, rapid professional interpretation of X-rays images is not always available, particularly in the resource-poor medical environment. Hence, the wide application of computer-aided diagnosis based on image classification has been urgently demanded. In this experiment, we simulate the medical application of our approach for diagnosing pediatric pneumonia by performing classification tasks on Chest X-rays dataset [74].

As shown in Fig. 7, this pediatric chest X-rays dataset is archived in three classes, including: 1) Normal; 2) Bacterial; and 3) Viral, and it has been preseparated as the training set (5216 samples) consisting of 2530 bacteria images, 1345 viral images and 1341 normal images, and testing set (624 samples) consisting of 242 bacteria images, 148 viral images and 234 normal images. Due to the fact of that the original image sizes are various from 529*968 to 952*1192, we have to resize the images into the uniform size of 224*224 via the

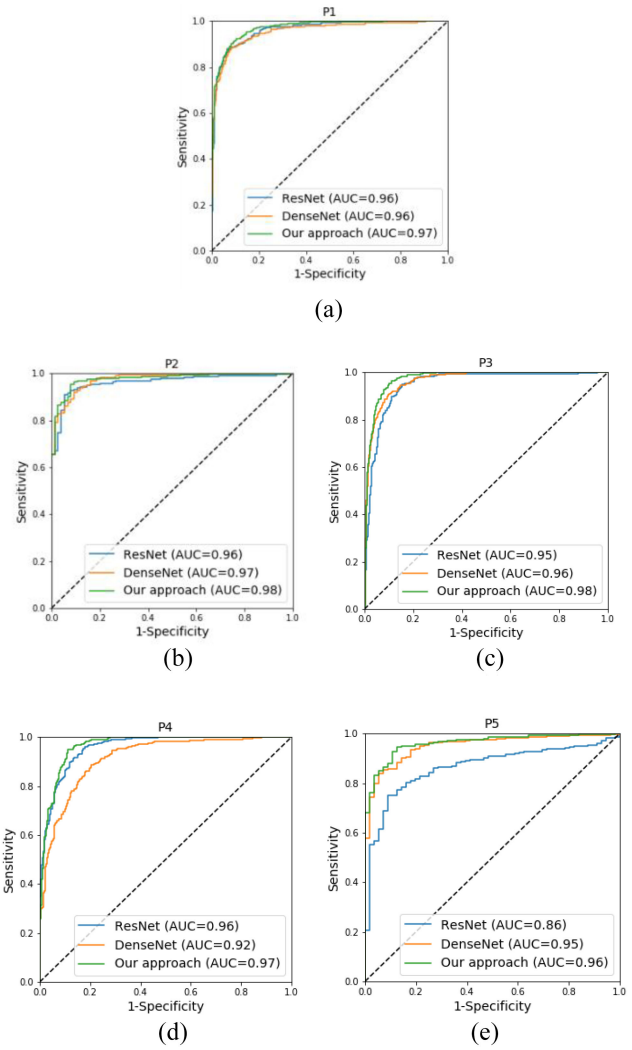


Fig. 6. ROC curves with AUC on the (a)–(e) P1–P5 test datasets (Our approach is GoogLeNet based).

(a) (b) (c)

Fig. 7. Chest X-rays dataset. (a) Normal. (b) Bacteria. (c) Viral.

bilinear interpolation method. For evaluating the multiclass classification performance of our approach, we further divided the predefined training dataset into training set (90%) and validation set (10%) randomly. For evaluating the binary classification performance of our approach, we construct three sets from this dataset. In Bacteria versus Viral set, the part of predefined training set, consisting of 2530 bacteria images and 1345 viral images, are randomly divided into training set (90%) and validation set (10%), and the corresponding testing set (390 samples) is obtained from the predefined testing set with bacterial (242 samples) and viral (148 samples) images. Meanwhile Bacteria versus Normal and Viral versus Normal sets are built in the same way. We also conduct the experiments for five trials, and report the average and standard deviation of experiment results obtained by different performance measures, respectively.

TABLE VIII
PERFORMANCES COMPARISON (%) OF THE COMPETING METHODS IN MULTICLASS CLASSIFICATION TASK ON CHEST X-RAYS DATASET

Methods	ACC	SEN	SPE	F1
AlexNet[22]	74.20±1.48	80.95±1.35	88.60±1.07	82.57±1.14
Net in Net[24]	77.40±0.91	86.67±0.94	89.92±0.88	86.95±0.95
VGG[23]	78.53±0.96	82.80±1.12	87.36±0.98	86.68±1.01
GoogLeNet[25]	79.65±0.91	84.62±0.94	89.65±1.53	86.34±1.47
ResNet[26]	81.25±0.98	85.90±0.95	89.26±1.14	86.40±1.05
DenseNet[27]	84.53±0.83	88.58±0.87	92.15±0.92	89.02±0.94
Our approach (NIN-based)	84.46±0.69	88.41±0.72	**97.11±0.83**	89.86±0.70
Our approach (GoogLeNet-based)	**86.70±0.82**	**89.34±0.91**	95.30±0.65	**91.26±0.87**

Table VIII lists all the results achieved in multiclass classification task on this dataset. It can be observed from the table that our approaches generally outperform other approaches, as our GoogLeNet approach has the best performance in terms of accuracy, sensitivity, and F1 score, and our NIN-based approach achieves the best result in terms of specificity. At the same time, the standard deviations demonstrate that our approach is more stable in comparison with others since they are insensitive to initial conditions. These results clearly demonstrate the outstanding performance of our approach for multiclass classification tasks on the Chest X-rays dataset.

By further evaluating the binary classification performance of our approaches on this dataset, it can be seen from Table IX that our approaches achieve the best results once again. A closer observation shows that our NIN-based approach is completely superior to the competing approaches in terms of sensitivity, specificity, and F1 score for Bacteria versus Viral classification task, while our GoogLeNet-based approach has the best classification accuracy, sensitivity, and F1 score on two binary classification tasks, and constantly has the best accuracy on all tasks. In practical diagnosis of pneumonia, discriminating the bacteria pneumonia patients from viral pneumonia patients is critical due to their different treatments. These results show that our approach provides a potential solution for assisting the diagnosis of pediatric pneumonia on X-rays images.

In order to make further comparison between our approach and state-of-the-art methods in practical medical applications, we further plot the ROC curves with AUC by implementing ResNet, DenseNet and our GoogLeNet-based approach on three binary classification tasks. As shown in Fig. 8, the curve obtained by our approach is closer to the optimal situation in the upper left corner than the ResNet and DenseNet methods, and its corresponding AUC area is larger than others. This significantly illustrates that our approach has greater potential for pediatric pneumonia diagnosis based on X-rays images.

D. Summary

The experiment results demonstrate that our approach is able to obtain high-quality classifiers with a simple algorithm

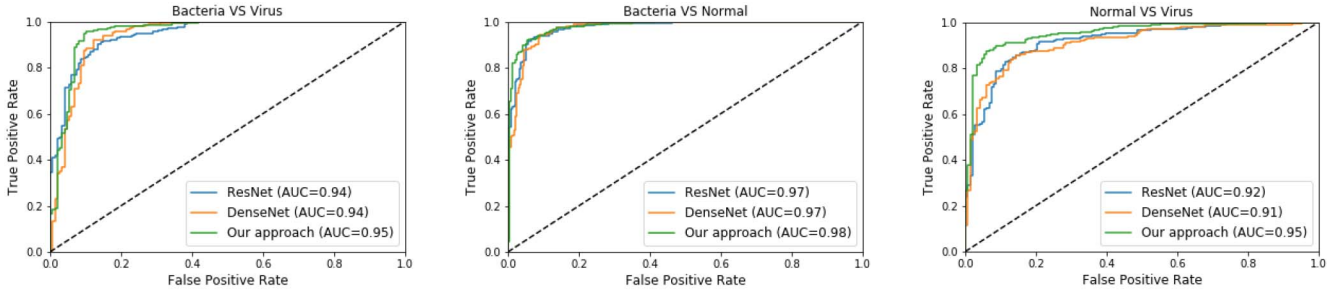


Fig. 8. ROC curves with AUC on three binary classification tasks (a) Bacteria VS Virus, (b) Bacteria VS Normal, and (c) Normal VS Virus listed in Table IX (Our approach is GoogLeNet based).

TABLE IX
PERFORMANCES COMPARISON (%) OF THE COMPETING METHODS IN BINARY CLASSIFICATION TASKS ON CHEST X-RAYS DATASET

Tasks	Measures	*AlexNet[22]*	*Net in Net[24]*	*VGG[23]*	*GoogLeNet[25]*	*ResNet[26]*	*DenseNet[27]*	***Our approach***	
								NIN-based	***GoogLeNet-based***
Bacteria VS Viral	ACC	84.62±1.14	90.51±0.97	89.17±1.05	90.51±0.88	87.38±0.89	91.74±0.83	94.23±0.81	**94.82±0.63**
	SEN	76.22±1.09	81.08±0.89	81.76±0.94	78.38±0.85	81.79±0.78	83.53±0.74	**84.03±0.87**	83.43±0.71
	SPE	95.87±0.87	96.28±0.82	96.28±0.92	97.93±0.79	96.69±0.77	90.91±0.68	**98.69±0.73**	98.17±0.66
	F1	76.56±0.96	86.64±0.93	84.05±1.03	86.25±0.86	89.14±0.90	89.67±0.82	**90.74±0.83**	89.78±0.60
Bacteria VS Normal	ACC	88.87±1.25	90.34±1.06	85.02±1.12	90.76±0.91	91.39±0.93	92.49±0.89	92.44±0.94	**94.91±0.87**
	SEN	94.21±1.44	89.26±1.13	87.92±1.36	95.04±0.84	93.39±0.81	95.93±0.82	94.63±0.91	**97.80±0.72**
	SPE	83.33±0.90	86.32±0.77	82.05±0.84	88.45±0.81	89.32±0.85	90.10±0.78	**90.17±0.74**	90.02±0.63
	F1	89.59±1.14	90.38±0.93	85.60±1.06	91.27±0.92	92.89±0.93	94.12±0.81	93.71±0.85	**95.00±0.78**
Viral VS Normal	ACC	83.51±0.74	83.77±0.81	84.82±0.79	86.91±0.71	84.82±0.72	84.29±0.68	89.01±0.55	**89.74±0.47**
	SEN	82.43±0.77	81.76±0.87	81.08±0.83	82.43±0.66	83.76±0.59	87.18±0.63	85.81±0.58	**86.78±0.51**
	SPE	84.19±0.70	87.04±0.75	87.18±0.80	88.74±0.61	84.49±0.69	87.18±0.59	**91.03±0.49**	90.88±0.43
	F1	79.48±0.62	79.61±0.68	80.54±0.81	82.99±0.54	84.82±0.79	83.45±0.76	85.81±0.37	**86.22±0.32**

implementation. Thus, it provides a promising yet easy-to-use technique for medical image classification tasks. As a result, the advantages of our approach can be summarized as follows.

First, we propose a novel training strategy called DTT to solve vanishing gradients problem. It has been justified in Section III-B, and tested on a variety of image datasets. As specifically shown in Fig. 2, our approach alleviates vanishing gradients occurred in deep neural networks to a certain extent, and gains a better performance on various image classification tasks on both standard benchmark and practical medical applications than their baselines and other state-of-the-art methods, which is shown in Tables IV and VI–IX. In addition, our approach intrinsically obtains multiple branch classifiers on different middle-level features, whereas ensemble learning is just right to provide an adequate solution for reconciling these branches of CNNs.

Second, we propose a two-stage selection scheme to select the optimal ensemble members from branch classifiers based on accuracy and diversity criteria, and further adopt weighting strategy to achieve the optimal ensemble classifier, which significantly prevents the problems of overfitting and local optima. This is validated by a benchmarking experiment from different perspectives. The results on benchmarking dataset presented in Section IV-B prove adequacy of the fundamental concepts introduced by our approach. Furthermore, two medical image datasets are used to evaluate the robustness and feasibility of our approach in the real-world medical applications. The results reported in Section IV-C showed the higher average values and lower standard deviations of various performance measures, which demonstrates that our approach can obviously reduce missed diagnosis rate and the misdiagnosis rate in practical diagnosis process. It indicates that with the help of our approach, a medical image-based CAD system could be built to diagnose patients efficiently and accurately.

V. CONCLUSION

In this article, we proposed a novel approach by combining the strengths of CNN and ensemble learning, and explored the feasibility of the approach to classify different types of medical images. In order to evaluate the performance of our approach on the classification tasks of medical image datasets, we designed several experiments, and compared our approach with several state-of-the-art CNN models. Experiment results showed that our approach can achieve superior performance on medical image classification tasks. This study has a great potential to lead to a new perspective for image-based CAD, and further facilitate more applications of CAD.

Although our findings show that our approach has excellent performance on medical image classification tasks, there are still some issues that have to be addressed in our future research. Our approach introduces many extra hyperparameters, such as the number of branches, the location of splitting, the number of selected base classifiers, and even the measurement of diversity. These hyperparameters bring the potential of further improvement of our proposed approach on medical image classification tasks. How to select a set of appropriate parameters for the target datasets with different forms and characteristics remains to be an interesting issue for future

studies. Moreover, inspired by the works [78], [79], we are going to explore the potential of incorporating visual attention mechanisms and different high-level feature into deep feature representations for improving the performance of medical image classification.

ACKNOWLEDGMENT

The authors are grateful to Alex Krizhevsky for collecting CIFAR-10 dataset, Paul Mooney for providing the BHI dataset on Kaggle Web site and Daniel Kermany, Kang Zhang, and Michael Goldbaum for publishing the Chest X-rays image dataset, which are used to conduct the image classification tasks in this article.

REFERENCES

[1] D. S. Kermany *et al.*, "Identifying medical diagnoses and treatable diseases by image-based deep learning," *Cell*, vol. 172, no. 5, pp. 1122–1131, 2018.

[2] J. Lotz *et al.*, "Patch-based nonlinear image registration for gigapixel whole slide images," *IEEE Trans. Biomed. Eng.*, vol. 63, no. 9, pp. 1812–1819, Sep. 2016.

[3] D. Zhang *et al.*, "Exploring task structure for brain tumor segmentation from multi-modality MR images," *IEEE Trans. Image Process.*, vol. 29, pp. 9032–9043, Sep. 2020.

[4] D. Zhang, G. Huang, Q. Zhang, J. Han, J. Han, and Y. Yu, "Cross-modality deep feature learning for brain tumor segmentation," *Pattern Recognit.*, vol. 110, Feb. 2021, Art. no. 107562.

[5] F. Ghesu *et al.*, "Multi-scale deep reinforcement learning for real-time 3D-landmark detection in CT scans," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 1, pp. 176–189, Jan. 2019.

[6] T. Tan, B. Platel, H. Huisman, C. I. Sanchez, R. Mus, and N. Karssemeijer, "Computer-aided lesion diagnosis in automated 3-D breast ultrasound using coronal spiculation," *IEEE Trans. Med. Imag.*, vol. 31, no. 5, pp. 1034–1042, May 2012.

[7] J. S. Duncan and N. Ayache, "Medical image analysis: Progress over two decades and the challenges ahead," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 85–106, Jan. 2000.

[8] B. V. Ginneken, B. M. T. H. Romeny, and M. A. Viergever, "Computer-aided diagnosis in chest radiography: A survey," *IEEE Trans. Med. Imag.*, vol. 20, no. 12, pp. 1228–1241, Dec. 2001.

[9] Y. Shi, Y. Gao, S. Liao, D. Zhang, Y. Gao, and D. Shen, "Semi-automatic segmentation of prostate in CT images via coupled feature representation and spatial-constrained transductive lasso," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 11, pp. 2286–2303, Nov. 2015.

[10] S. Singh, J. Maxwell, J. A. Baker, J. L. Nicholas, and J. Y. Lo, "Computer-aided classification of breast masses: Performance and interobserver variability of expert radiologists versus residents," *Radiology*, vol. 258, no. 1, pp. 73–80, 2011.

[11] J. Segyeong, Y. Y. Seok, M. W. Kyung, and K. H. Chan, "Computer-aided diagnosis of solid breast nodules: Use of an artificial neural network based on multiple sonographic features," *IEEE Trans. Med. Imag.*, vol. 23, no. 10, pp. 1292–1300, Oct. 2004.

[12] M. B. McCarville *et al.*, "Distinguishing benign from malignant pulmonary nodules with helical chest CT in children with malignant solid tumors," *Radiology*, vol. 239, no. 2, pp. 514–520, 2006.

[13] S. Yang, C. Weidong, Z. Yun, and F. D. Dagan, "Feature-based image patch approximation for lung tissue classification," *IEEE Trans. Med. Imag.*, vol. 32, no. 4, pp. 797–808, Apr. 2013.

[14] F. Zhang *et al.*, "Lung nodule classification with multilevel patch-based context analysis," *IEEE Trans. Biomed. Eng.*, vol. 61, no. 4, pp. 1155–1166, Apr. 2014.

[15] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[16] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[17] L. Nanni, A. Lumini, and S. Brahnam, "Local binary patterns variants as texture descriptors for medical image analysis," *Artif. Intell. Med.*, vol. 49, no. 2, pp. 117–125, 2010.

[18] H.-I. Suk, S.-W. Lee, and D. Shen, "Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis," *NeuroImage*, vol. 101, pp. 569–582, Jul. 2014.

[19] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[20] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[21] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.

[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014. [Online]. Available: arXiv:1409.1556

[24] H. Shin, M. R. Orton, D. J. Collins, S. J. Doran, and M. O. Leach, "Stacked autoencoders for unsupervised feature learning and multiple organ detection in a pilot study using 4D patient data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1930–1943, Aug. 2013.

[25] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2015, pp. 1–9.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. Comput. Vis. ECCV*, 2016, pp. 630–645.

[27] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 2261–2269.

[28] G. Huang, S. Liu, L. V. D. Maaten, and K. Q. Weinberger, "CondenseNet: An efficient DenseNet using learned group convolutions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2752–2761.

[29] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.

[30] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *Int. J. Uncertainty Fuzziness Knowl. Based Syst.*, vol. 6, no. 2, pp. 107–116, 1998.

[31] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," 2012. [Online]. Available: arXiv:1211.5063

[32] Y. N. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio, "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization," *Proc. NIPS*, 2014, pp. 2933–2941.

[33] K. Grm, V. Štruc, A. Artiges, M. Caron, and H. K. Ekenel, "Strengths and weaknesses of deep learning models for face recognition against image degradations," *IET Biometr.*, vol. 7, no. 1, pp. 81–89, 2018.

[34] H. Jeelani, J. Martin, F. Vasquez, M. Salerno, and D. S. Weller, "Image quality affects deep learning reconstruction of MRI," in *Proc. IEEE 15th Int. Symp. Biomed. Imag. (ISBI)*, 2018, pp. 357–360.

[35] Y. Yang and K. Chen, "Temporal data clustering via weighted clustering ensemble with different representations," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 2, pp. 307–320, Feb. 2011.

[36] Y. Yang and K. Chen, "Time series clustering via RPCL network ensemble with different representations," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 41, no. 2, pp. 190–199, Mar. 2011.

[37] J. Arevalo, F. A. González, R. Ramos-Pollán, J. L. Oliveira, and M. A. Guevara Lopez, "Representation learning for mammography mass lesion classification with convolutional neural networks," *Comput. Methods Progr. Biomed.*, vol. 127, pp. 248–257, Apr. 2016.

[38] A. Cruzroa, H. Gilmore, M. Feldman, J. Tomaszewski, and A. Madabhushi, "Automatic detection of invasive ductal carcinoma in whole slide images with convolutional neural networks," in *Proc. SPIE Int. Soc. Opt. Eng.*, vol. 9041, 2014, pp. 139–144.

[39] R. Rasti, H. Rabbani, A. Mehridehnavi, and F. Hajizadeh, "Macular OCT classification using a multi-scale convolutional neural network ensemble," *IEEE Trans. Med. Imag.*, vol. 37, no. 4, pp. 1024–1034, Apr. 2018.

[40] G. Carneiro and J. C. Nascimento, "Combining multiple dynamic models and deep learning architectures for tracking the left ventricle endocardium in ultrasound data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2592–2607, Nov. 2013.

[41] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Proc. NIPS*, 2006, pp. 153–160.

[42] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," 2013. [Online]. Available: arXiv:1302.4389

[43] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 1026–1034.

[44] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. ICML*, 2010, pp. 807–814.

[45] C.-Y. Lee, S. Xie, P. W. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," 2014. [Online]. Available: arXiv:1409.5185

[46] D. Yu, F. Seide, G. Li, and L. Deng, "Exploiting sparseness in deep neural networks for large vocabulary speech recognition," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2012, pp. 4409–4412.
[47] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," 2012. [Online]. Available: arXiv:1207.0580
[48] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015. [Online]. Available: arXiv:1502.03167
[49] G. Kang, J. Li, and D. Tao, "ShakeOut: A new regularized deep neural network training scheme," in *Proc. 13th AAAI Conf. Artif. Intell.*, 2016, pp. 1751–1757.
[50] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1319–1327.
[51] C. Zhang and Y. Ma, *Ensemble Machine Learning: Methods and Applications*. Heidelberg, Germany: Springer, 2012.
[52] Z. H. Zhou, *Ensemble Methods: Foundations and Algorithms*. London, U.K.: Taylor & Francis, 2012, p. 236.
[53] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, 1996.
[54] Y. Freund, "Boosting a weak learning algorithm by majority," *Inf. Comput.*, vol. 121, no. 2, pp. 256–285, 1995.
[55] D. H. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, no. 2, pp. 241–259, 1992.
[56] P. Yang, P. D. Yoo, J. Fernando, B. B. Zhou, Z. Zhang, and A. Y. Zomaya, "Sample subset optimization techniques for imbalanced and ensemble learning problems in bioinformatics applications," *IEEE Trans. Cybern.*, vol. 44, no. 3, pp. 445–455, Mar. 2014.
[57] S. Pan, J. Wu, X. Zhu, and C. Zhang, "Graph ensemble boosting for imbalanced noisy graph stream classification," *IEEE Trans. Cybern.*, vol. 45, no. 5, pp. 954–968, May 2015.
[58] Z. Yu *et al.*, "Hybrid incremental ensemble learning for noisy real-world data classification," *IEEE Trans. Cybern.*, vol. 49, no. 2, pp. 403–416, Feb. 2019.
[59] Z. Yu, L. Li, J. Liu, and G. Han, "Hybrid adaptive classifier ensemble," *IEEE Trans. Cybern.*, vol. 45, no. 2, pp. 177–190, Feb. 2015.
[60] Y. Yang and J. Jiang, "Hybrid sampling-based clustering ensemble with global and local constitutions," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 5, pp. 952–965, May 2016.
[61] Y. Yang and J. Jiang, "Adaptive bi-weighting toward automatic initialization and model selection for HMM-based hybrid meta-clustering ensembles," *IEEE Trans. Cybern.*, vol. 49, no. 5, pp. 1657–1668, May 2019.
[62] Z. Yu *et al.*, "Adaptive semi-supervised classifier ensemble for high dimensional data classification," *IEEE Trans. Cybern.*, vol. 49, no. 2, pp. 366–379, Feb. 2019.
[63] A. Kumar, J. Kim, D. Lyndon, M. Fulham, and D. Feng, "An ensemble of fine-tuned convolutional neural networks for medical image classification," *IEEE J. Biomed. Health Informat.*, vol. 21, no. 1, pp. 31–40, Jan. 2017.
[64] F. Xie, H. Fan, Y. Li, Z. Jiang, R. Meng, and A. Bovik, "Melanoma classification on dermoscopy images using a neural network ensemble model," *IEEE Trans. Med. Imag.*, vol. 36, no. 3, pp. 849–858, Mar. 2017.
[65] M. M. Fraz *et al.*, "An ensemble classification-based approach applied to retinal blood vessel segmentation," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 9, pp. 2538–2548, Sep. 2012.
[66] R. A. Ochs *et al.*, "Automated classification of lung bronchovascular anatomy in CT using AdaBoost," *Med. Image Anal.*, vol. 11, no. 3, pp. 315–324, 2007.
[67] M. Termenon and M. Graña, "A two stage sequential ensemble applied to the classification of Alzheimer's disease based on MRI features," *Neural Process. Lett.*, vol. 35, no. 1, pp. 1–12, 2012.
[68] B. Yoshua, *Learning Deep Architectures for AI* (Learning Deep Architectures for AI). London, U.K.: Now Found. Trends, 2009, p. 136.
[69] C. A. Shipp and L. I. Kuncheva, "Relationships between combination methods and measures of diversity in combining classifiers," *Inf. Fusion*, vol. 3, no. 2, pp. 135–148, 2002.
[70] G. Giacinto and F. Roli, "Design of effective neural network ensembles for image classification purposes," *Image Vis. Comput.*, vol. 19, no. 9, pp. 699–707, 2001.
[71] G. Martínez-Muñoz, D. Hernández-Lobato, and A. Suárez, "An analysis of ensemble pruning techniques based on ordered aggregation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 245–259, Feb. 2009.
[72] (2009). *CIFAR-10*. [Online]. Available: http://www.cs.toronto.edu/~kriz/cifar.html
[73] (2017). *Breast Histopathology Images*. Available: https://www.kaggle.com/paultimothymooney/breast-histopathology-images
[74] (2018). *Large Dataset of Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images*. doi: 10.17632/rscbjbr9sj.3.
[75] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014. [Online]. Available: arXiv:1412.6980
[76] C. DeSantis, R. Siegel, P. Bandi, and A. Jemal, "Breast cancer statistics, 2011," *CA Cancer J. Clin.*, vol. 61, no. 6, pp. 409–418, 2011.
[77] R. A. Adegbola, "Childhood Pneumonia as a Global Health Priority and the Strategic Interest of The Bill & Melinda Gates Foundation," *Clin. Infectious Diseases*, vol. 54, no. s2, pp. S89–S92, 2012.
[78] J. Han, K. N. Ngan, M. Li, and H.-J. Zhang, "Unsupervised extraction of visual attention objects in color images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 1, pp. 141–145, Jan. 2006.
[79] J. Han *et al.*, "Representing and retrieving video shots in human-centric brain imaging space," *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2723–2736, Jul. 2013.

Yun Yang received the B.Sc. degree (Hons.) in information technology and telecommunication from Lancaster University, Lancaster, U.K., in 2004, the M.Sc. degree in advanced computing from Bristol University, Bristol, U.K., in 2005, the M.Phil. degree in informatics, and the Ph.D. degree in computer science from the University of Manchester, Manchester, U.K., in 2006 and 2011, respectively.

He was a Research Fellow with the University of Surrey, Guildford, U.K., from 2012 to 2013. He is currently with the National Pilot School of Software, Yunnan University, Kunming, China, as a Full Professor of Machine Learning, and the Director of Yunnan Education Department Key Laboratory of Data Science and Intelligent Computing and Kunming Key Laboratory of Data Science and Intelligent Computing. His current research interests include machine learning, data mining, pattern recognition and temporal data process and analysis.

Prof. Yang serves as an Associate Editor for *Neural Networks* and *Complex & Intelligent Systems*.

Yuanyuan Hu received the B.Sc. and M.Sc. degrees in software engineering from Yunnan University, Kunming, China, in 2017 and 2020, respectively.

Her current research interests include deep learning and image data process and analysis.

Xingyi Zhang (Senior Member, IEEE) received the B.Sc. degree from Fuyang Normal College, Fuyang, China, in 2003, and the M.Sc. and Ph.D. degrees from the Huazhong University of Science and Technology, Wuhan, China, in 2006 and 2009, respectively.

He is currently a Professor with the School of Computer Science and Technology, Anhui University, Hefei, China. His current research interests include unconventional models and algorithms of computation, multiobjective optimization, and membrane computing.

Prof. Zhang is a recipient of the 2018 IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Outstanding Paper Award.

Song Wang (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana–Champaign (UIUC), Champaign, IL, USA, in 2002.

He was a Research Assistant with the Image Formation and Processing Group, Beckman Institute, UIUC from 1998 to 2002. In 2002, he joined the Department of Computer Science and Engineering, University of South Carolina, Columbia, SC, USA, where he is currently a Professor. His current research interests include computer vision, image processing, and machine learning.

Prof. Wang is currently serving as the Publicity/Web Portal Chair of the Technical Committee of Pattern Analysis and Machine Intelligence of the IEEE Computer Society, an Associate Editor of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, *Pattern Recognition Letters*, and *Electronics Letters*. He is a member of IEEE Computer Society.