IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 23, 2019, accepted July 31, 2019, date of publication August 5, 2019, date of current version August 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2933206

Brain Image Recognition Algorithm and High Performance Computing of Internet of Medical Things Based on Convolutional Neural Network

YUXI LIU[1] **AND JUN XIONG**[2]
[1]Computing and IT College of Sciences and Engineering, University of Tasmania, Hobart, TAS 7000, Australia
[2]Faculty of Engineering and Information Technologies, The University of Sydney, Sydney, NSW 2006, Australia

Corresponding author: Yuxi Liu (yuxil@utas.edu.au)

ABSTRACT Due to the wide variety of medical images and the complexity of the human body structure, the characteristics of manual extraction of medical images are difficult, the adaptive ability is poor, and the classification effect needs to be improved. Aiming at the shortcomings of traditional medical image recognition methods, this paper proposes an adaptive convolutional neural network model CNN-BN-PReLU based on the convolutional neural network method. The model first performs batch normalization (BN) processing on the input of each feature map of each layer of network, and then adaptively adjusts the parameters by using Parametric Rectified Linear Unit (PReLU) to compare the BN algorithm. Based on the performance before and after the activation function, an adaptive convolutional neural network model is constructed. The experimental results show that the model can abstract the image features without artificial intervention, speed up the network convergence speed and shorten the training time, and significantly improve the image recognition rate and reduce the misdiagnosis rate and missed diagnosis rate of the disease.

INDEX TERMS Brain image recognition, high performance computing, convolutional neural network.

I. INTRODUCTION

Every aspect of human life contains shadows of images. With the continuous development of science and technology, many researchers have done a lot of research on digital images, and applied them to actual production and life, and constantly improve people's lives. Now digital images the development of processing technology is at a new stage. Medical image processing technology uses medical images as information carriers to establish and optimize mathematical models as solutions to practical medical problems and life. Medical image pattern recognition technology is an important part of medical image processing. Its main function is to analyze the medical information in the image by computer's high-speed computing ability, automatically process some important medical information, and then assist the doctor to complete the classification of certain diseases to identify the problem. Convolutional neural network method is an emerging technology in the field of pattern recognition that imitates the automatic learning process of human brain. It is developed by researchers for early neural network research and has been successfully applied to various types of image processing problems [1]–[3], has a strong adaptability. Combined with the problems encountered in brain image recognition research, it is of great significance to carry out the application of convolutional neural network in brain image classification and recognition [3].

In recent years, brain image recognition technology has played a good role in the analysis and processing of medical images, and has been gradually applied to the analysis and diagnosis of brain diseases, making great contributions to medical and medical undertakings. Shen uses rough sets for patient basic information processing [4]. Bourouis *et al.* used image processing technology and an artificial neural network-based image classifier to automatically analyze retinal images and classify images according to disease conditions [5], [6]. Chaves *et al.* applied artificial neural networks to brain images, using several pattern classifiers, including K-nearest neighbors, classification trees, support vector machines, and templates combined with feedforward

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Wei.

neural networks for several coronary slice features of interest. Research is conducted to analyze the condition of patients with Alzheimer's disease, which will help early identification of Alzheimer's disease [7]. Wang *et al.* used support vector machines for the efficient classification of X-ray brain images to aid in the diagnosis and treatment of diseases [8].

With the continuous development of brain image theory and the increasing demand for brain images, feature extraction, multi-classifier combination and distributed recognition systems have emerged. Avci *et al.* proposed a new technique based on adaptive wavelet entropy energy and neural network classifier for brain cell recognition of microscopic image rotation and scaling. Discrete wavelet transform and adaptive wavelet entropy and energy were used in the feature extraction phase. Adaptive feature extraction enhances the excellent characteristics of artificial neural network classifiers [9]. Paker *et al.* used the attribute-weighted K-center clustering algorithm to preprocess the brain image data, and then use the support vector machine for analysis and identification [10]. Medeiros *et al.* established a fuzzy temporal rule classifier based on fuzzy rough sets and temporal logic, which achieved good results [11]. Haghshenas *et al.* classify X-ray brain images by radial basis network classifier, and use simulated annealing algorithm to avoid the network falling into local optimum and optimize the adaptive network structure [12]. Zheng applied fuzzy rough sets to gene selection in gene expression profiling [13]. In the study of multimodal brain images, Tian used the support vector machine to train and test its boundary features, radius and area ratio [14]. When Su *et al.* studied the brain magnetic resonance image of human brain, the brain image was first transformed from the spatial domain to the frequency domain, and then the texture feature extraction was performed. The support vector machine was used to distinguish between the lesion and the non-lesion image [15]. Hao *et al.* have some redundancy in the diagnosis of cancer symptom data, and the rough set theory is used to simplify the data attributes and improve the accuracy of cancer symptom classification [16].

The quality of feature selection directly affects the final result of classification recognition. Therefore, it is very important to select good features. However, in the traditional method, the selection of features is based on artificial, the process of manually selecting features is more complicated, and the obtained results are unstable and the adaptability is not strong. The convolutional neural network theory breaks through the bottleneck of manual selection features, and continuously surpasses the traditional recognition algorithm in classification and recognition. The successful application in brain image classification and recognition has caused great response and provided new research for medical image classification and recognition research vitality. In 1995, David Field and others demonstrated that some basic structural features can combine a wide variety of complex structures [17]. As shown in Figure 1, an image can be decomposed into 64 basic structures, which are combined with several texture structures. According to the different weights of the infrastructure, they can be combined into a new image representation.

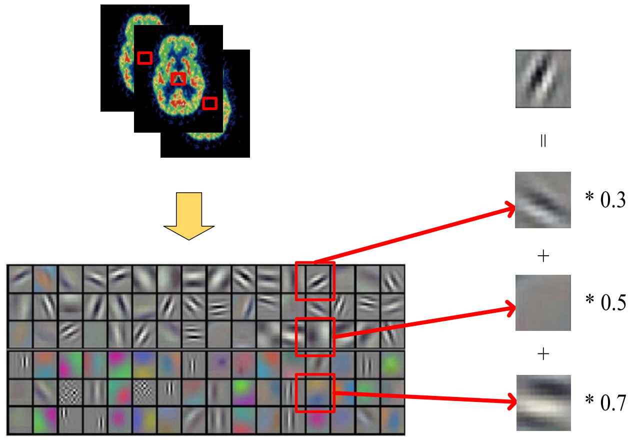

FIGURE 1. **Basic composition granularity of pictures.**

Aiming at the cumbersome process of traditional brain image recognition algorithm and difficulty in feature extraction, this chapter proposes an adaptive convolutional neural network model. This model is not a convolutional neural network in the past. Instead, it uses a batch normalization algorithm to speed up the training. A parametric linear correction unit is used to improve the classification accuracy. The model firstly performs batch normalization on the input of each feature map of each layer of network, and then adaptively adjusts the parameters by using the parameterized linear correction unit to compare the performance of the batch normalization algorithm before and after the activation function. Adapt to the convolutional neural network model. Compared with the two classical convolutional neural network models, the model not only accelerates the network convergence speed, shortens the training time, but also significantly improves the image recognition rate and reduces the rate of misdiagnosis and missed diagnosis.

Specifically, the technical contributions of our paper can be concluded as follows:

First: In this paper, the research status of deep learning in brain images and the challenges and deficiencies of traditional brain image recognition methods are described and summarized in detail. The related methods and models of convolutional neural network are studied. The network layers and the number of convolution kernels are also discussed. The structural parameters are systematically studied and analyzed, which provides reference value for the construction of convolutional neural networks.

Second: An adaptive convolutional neural network model is proposed. The model not only speeds up the network convergence speed, shortens the training time, but also significantly improves the image recognition rate and reduces the rate of misdiagnosis and missed diagnosis.

The rest of our paper was organized as follows. Theory of convolutional neural network was introduced in Section II. Section III described related theoretical derivation of the

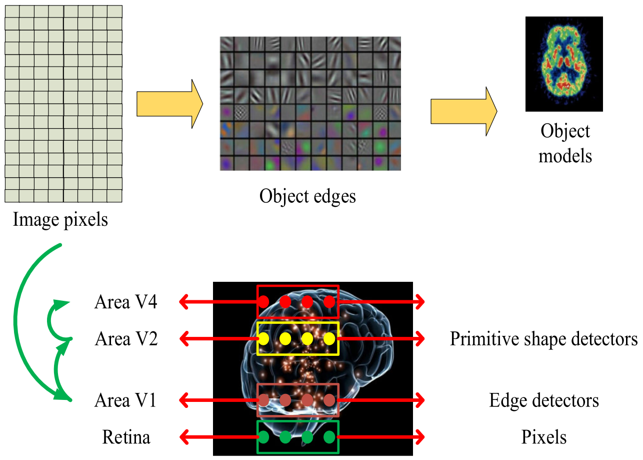

FIGURE 2. The basic idea of convolutional neural network.

proposed system. Experimental results and analysis were discussed in detail in Section IV. Finally, Section V concluded the whole paper.

II. THEORY OF CONVOLUTIONAL NEURAL NETWORK

A. CONVOLUTIONAL NEURAL NETWORK FOUNDATION

Convolutional neural network is a cross-cutting field of research in neural networks, artificial intelligence, graphical modeling, optimization, pattern recognition, and signal processing. There are many similar definitions and high-level descriptions of convolutional neural network. According to the relevant explanations in [18], it can be summarized as follows: Convolutional neural network is a subfield of machine learning, which attempts to transform data through a series of multi-level nonlinear transformations perform an abstract algorithm.

Convolutional neural network is a multi-level machine learning algorithm based on characterizing learning for complex relationships between simulated data. An observation can be represented in a variety of ways, such as using a matrix of intensity values to represent pixels. Some representations make it easier for the algorithm to complete the learning task. The goal of characterizing learning is to find better representations and build better models to learn these representations.

Convolutional neural network establishes a hierarchical model structure similar to the human brain by simulating the brain system with rich hierarchical structure, and extracts the input data step by step to form a more abstract high-level representation. Convolutional neural network uses multi-layer nonlinear information processing to achieve supervised or unsupervised feature extraction and transformation, pattern analysis and classification to explain data such as images, sounds, and text. High-level features and concepts are defined by lower-level features and concepts, and the same low-level concepts can be used to define many high-level concepts [19]. Such a hierarchical structure is called a deep structure.

Figure 2 shows the basic idea of convolutional neural network. As can be seen from Figure 2, the human body introduces the received signal into the V1 region of the brain by ingesting pixel-level signals, where V1 is the first region in the brain that performs significant advanced processing on the visual input, through the V1 region. Extract the edge features, then pass the processed signals to the V2 area, abstract the output of the previous stage, extract the high-order features such as the shape of the object, and finally extract the higher-level visual abstraction through the V4 area until all The information features are identified. In general, the artificial neural network is a hierarchical structure that mimics the cerebral cortex, and gradually extracts features from the lower layer to the upper layer to discover the distributed feature representation of the data.

There are two main aspects in the various high-level descriptions of convolutional neural network mentioned above:

(1) The model consists of multi-level or stage nonlinear information processing;

(2) Supervised or unsupervised learning of feature representation, more abstract as the number of layers increases.

Convolutional neural network uses layered abstraction, and high-level concepts are learned through low-level concepts. This hierarchical structure is usually constructed using a greedy layer-by-layer training algorithm, and selects effective features that contribute to machine learning. Many convolutional neural network algorithms are presented in the form of unsupervised learning, so these algorithms can be applied in unlabeled data that other algorithms cannot match. This type of data is richer and more accessible than tagged data, which is an important advantage of convolutional neural network [20].

Today, three important reasons for the popularity of convolutional neural network are: first, the processing power of the chip has been greatly improved, for example, the emergence of general-purpose graphics processing units, and secondly, the cost of hardware computing is significantly reduced, and third, machine learning and signal/information processing in research. These advances have enabled convolutional neural network methods to efficiently utilize complex, integrated nonlinear functions to learn distributed and hierarchical feature representations and to make efficient use of tagged and unlabeled data.

A deep network is a network with multiple hidden layer structures. By introducing a deep network, this paper can implement a complex nonlinear function by learning a deep nonlinear network to calculate more complex input features [21]. Since each hidden layer can nonlinearly transform the output of the upper layer, the deep network has better expressive power than the shallow network. For example, it can learn more complicated functional relationships and exhibits the ability to learn the essential characteristics of data in a few samples. The main advantage of deep networks is that they can represent a much larger set of functions in a much simpler way than traditional shallow networks, and the advantage of multiple layers is that they can use fewer parameters to represent complex functional relationships. As shown in the

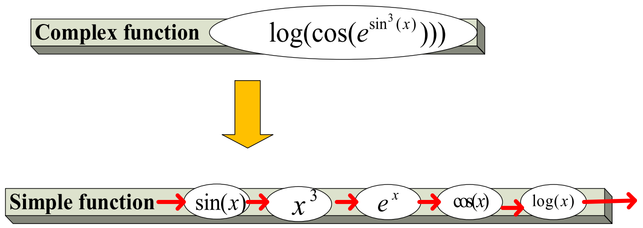

FIGURE 3. Represent the complex function use multilayer structure.

Figure, to express a complex function $\log(\cos(e^{\sin^3(x)}))$, it is difficult to express it succinctly with a traditional single-layer structure, and with a deep structure of multiple hidden layers, it is possible to represent more complex functions with fewer parameters, using multiple layers. The simple structure $\sin(x)$, x^3, e^x, $\cos(x)$, $\log(x)$ to represent the above complex functions is much easier.

From the perspective of the hierarchical structure of machine learning models, there are two types of ML: Shallow Learning (SL) and convolutional neural network.

SL refers to an ML model that contains only one or two layers of nonlinear feature transformations [22]. Shallow learning models include k-means clustering, SVM, logistic regression, maximum entropy, and so on. Because of their limited ability to represent, they can only solve some simple practical problems. When there are complex real-world problems, they are not well expressed. In general, the more network layers a model has, the more it can extract high-dimensional features, which means it can perform more complex tasks. However, complex models with many network levels also have problems such as low training efficiency and easy to fall into over-fitting. Thanks to the rapid development of cloud computing and big data, both computing power and training data have increased significantly, which not only eases training inefficiency, but also reduces the risk of overfitting. Therefore, the complex model represented by convolutional neural network has gradually become a hot spot for researchers. A typical convolutional neural network model is a neural network with more network layers. For the neural network model, the methods to improve the complexity and performance of the network mainly include increasing the number of hidden layers and increasing the number of hidden neurons. As the number of hidden layers increases, the corresponding parameters such as neuron weights and thresholds will be more. However, from the perspective of improving the complexity of the model, the latter method is far less effective than the former, because the former can not only increase the number of neurons with activation functions, but also increase the number of layers in which the activation function is nested [23]. Convolutional neural network training process: greedy layer-by-layer unsupervised pre-training is an effective means of DL model training. The basic idea is that each layer uses unsupervised learning pre-training, taking the output of the previous layer as input and outputting a new representation of the data. After the pre-training is completed, the joint training algorithm is fine-tuned. A summary and summary of the above is shown in Table 1.

TABLE 1. Shallow learning and convolutional neural network.

	Shallow learning	Convolutional neural network
Sample size	Small sample	Large sample
Learning method	Supervised learning	Unsupervised learning
Number of network layers	One or two layers with less layers	More
Extract feature dimension	Low dimensional feature	High dimensional feature
Model complexity	Lower	Higher
Training efficiency	Lower	Higher
Applicable task	Simple task	Complex task

B. CONVOLUTIONAL NEURAL NETWORK FOUNDATION

1) RESTRICTED BOLTZMANN MACHINE (RBM)

Restricted Boltzmann Machine is a typical generation model in convolutional neural network, including a layer of visible variables and single-layer hidden variables. RBM is essentially an undirected probability graph model that can be stacked to form a deeper model with no connections allowed between any of the visible or hidden layers [24]. In general, the visible layer is used to describe one aspect or feature of the data, and the hidden layer can be viewed as a feature extraction layer. This chapter only introduces the binary version of the restricted Boltzmann machine.

The standard RBM is an energy-based model with binary visible and hidden elements. Let the visible layer consist of a set of n_v binary random variables, collectively referred to as v; the hidden layer of n_h binary random variables is denoted as h. The joint probability distribution of RBM is specified by the energy function:

$$P(\mathbf{v} = v, \mathbf{h} = h) = e^{-E(v,h)}/Z \tag{1}$$

$$E(v, h) = -b^T v - c^T - v^T Wh \tag{2}$$

where b, c, and W are unconstrained, real-valued learnable parameters, and Z in equation (1) is a normalized constant called a partition function:

$$Z = \sum_v \sum_h e^{-E(v,h)} \tag{3}$$

As can be seen from the above formula, the model is divided into two groups of v and h, and the interaction between them is described by the matrix W. Figure 4 depicts the RBM model in a bipartite graph structure with one layer visible and one hidden layer. Unlike the Boltzmann machine, there is no connection between any two visible elements in the RBM visible layer, as is the hidden layer.

For a binary restricted Boltzmann machine, the conditional probability distributions for v and h are given by the following

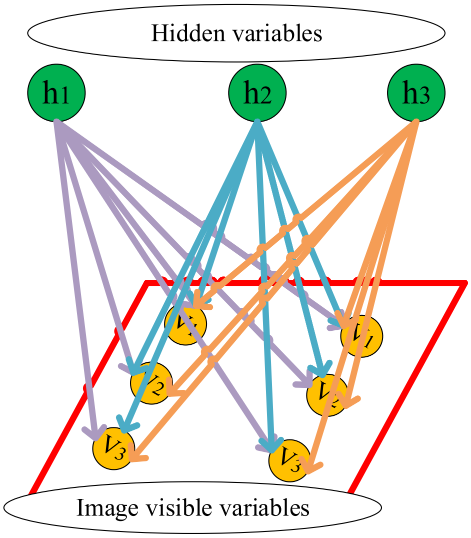

FIGURE 4. RBM drawn as a Markov network.

formula:

$$p(h_i = 1|\mathbf{v}) = \sigma(\mathbf{v}^T W_{:,i} + b_i) \tag{4}$$

$$p(h_i = 0|\mathbf{v}) = 1 - \sigma(\mathbf{v}^T W_{:,i} + b_i) \tag{5}$$

It can alternate between sampling all h at the same time and sampling all v at the same time with combining these properties results in efficient Gibbs sampling. Since the energy function is essentially a linear function, its derivative is extremely easy to obtain. Such as,

$$\partial/\partial W_{i,j} E(\mathbf{v}, \mathbf{h}) = -\mathbf{v}_i \mathbf{h}_j \tag{6}$$

The RBM can be successfully trained by Gibbs sampling and derivative calculation. The training algorithm often uses the Contrastive Divergence algorithm [25]. The training model can get the representation h of the data v, using $E_{h\sim p(\mathbf{h}|\mathbf{v})}[h]$ as a set of features describing v.

In summary, RBM uses a multi-layer hidden variable to perform representational learning with efficient interactions between matrix parameterizations, making it capable of rapid learning.

2) DEEP BELIEF NETWORK (DBN)

The Deep Belief Network (DBN) was one of the first non-convolution models to successfully apply deep architecture training.

In 2006, the DBN proposal marked the revival of current convolutional neural network. It outperformed the kernel SVM on the MNIST dataset, overcoming the problem that artificial neural networks are prone to local optimality and difficult to train, and can also utilize a large number of unlabeled The unsupervised learning of the samples shows that DBN has a wide range of application feasibility. At present, based on DBN, many new convolutional neural network models have emerged, such as convolution belief networks.

The deep belief network uses the generation model in the pre-training process and the back propagation in the fine-tuning phase. It is a fast learning algorithm that can find the best parameters. DBN is a hybrid graph model involving directed and undirected connections. In DBN, each layer is a constrained Boltzmann machine, i.e. the entire network can be viewed as a number of RBM stacks. As with RBM, there are no intra-layer connections in the DBN; the difference is that there are multiple hidden layers in the DBN, where hidden units in the hidden layer are usually binary.

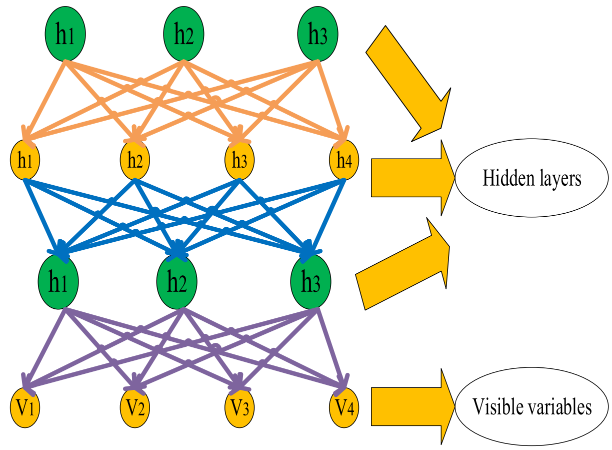

FIGURE 5. Deep belief network model.

Figure 5 is a schematic diagram of the general model of the deep belief network. As can be seen from the model, the top two layers are hidden layers and the bottom layer is the visible layer. An undirected connection is taken between the two hidden layers, and a directed connection is taken between the hidden layer and the visible layer, and the arrow points to the v layer closest to the data.

A DBN with 1 hidden layer consists of 1 weight matrix, $W^{(1)}, \ldots, W^{(l)}$ contains 1+1 offset vectors. $b^{(0)}, \ldots, b^{(l)}$, where $b^{(0)}$ is the offset of the visible layer. The probability distribution represented by DBN is given by:

$$P(h^{(l)}, h^{(l-1)}) \propto e^{(b^{(l)^T} h^{(l)} + b^{(l-1)^T} h^{(l-1)} + b^{(l-1)^T} W^{(l)} h^{(l)})} \tag{7}$$

$$P(h_i^{(k)} = 1, h^{(k+1)}) = \sigma(b_i^{(k)} + W_{:,i} h^{(k+1)}) \tag{8}$$

$$P(v_i = 1, h^{(1)}) = \sigma(b_i^{(0)} + W_{:,i}^{(1)^T} h^{(1)}) \tag{9}$$

DBN training methods can use CD algorithm or random maximum likelihood method to maximize $E_{v\sim pdata} \log p(v)$. The parameters of the RBM define the parameters of the first layer of the DBN. Then, the second RBM training is approximately maximized:

$$E_{v\sim pdata} E_{h(1)\sim p(1)(h^{(1)}|\mathbf{v})} \log p^{(2)}(h^{(1)}) \tag{10}$$

where $p^{(1)}$ is the probability distribution of the first RBM representation and $p^{(2)}$ is the probability distribution of the second RBM representation. That is, the first RBM is driven by data, and the second RBM is determined based on the distribution of the first RBM hidden unit sampling definition, which is derived backwards. Through this process, you can

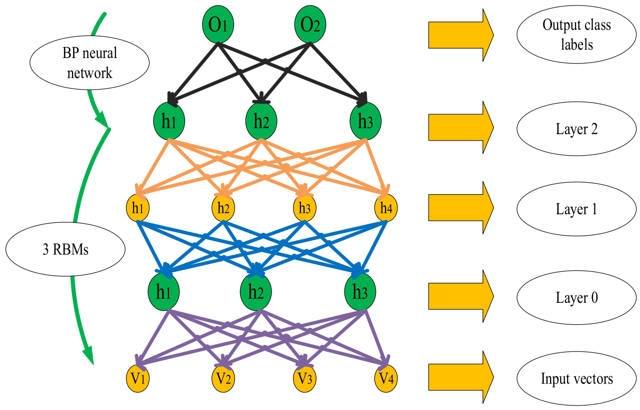

FIGURE 6. DBN for sample classification diagram.

add any number of layers to DBM. A new RBM is a model for the previous RBM.

Figure 6 is a schematic diagram of DBN for sample classification. The model in the Figure consists of 1 input layer, 1 output layer and 3 hidden layers. The classification purpose is achieved by inputting the sample matrix into the input layer and finally outputting the classification label.

In practice, a trained DBN can be used directly as a model to obtain weights and offsets, and use these parameters to define a BP neural network:

$$h^{(1)} = \sigma(b^{(1)} + v^T W^{(1)}) \tag{11}$$

After initializing the BP network using the weights and offsets obtained after the DBN's generation training, the network is trained to perform the classification task. Compared with traditional neural networks, deep belief networks have improved in the pre-training and fine-tuning phases. In recent years, the deep belief network has been widely used in the field of image recognition, which greatly improves the recognition accuracy of images.

3) DEEP BOLTZMANN MACHINE (DBM)

The Deep Boltzmann Machine (DBM) is another depth-generating model proposed by Hinton [26]. Like RBM and DBN, DBM is also an energy-based model, which usually only contains binary units, but can easily be extended to real-valued visible units. DBM has something in common with DBN and RBM, but it also has its own differences:

1) Unlike DBN, it is a model with only undirected connections;

2) Unlike RBM, DBM has multiple hidden variables, while RBM has only one layer.

Figure 7 is a schematic diagram of the structure of the DBM general model. The model is the same as the DBN model in Figure 5, consisting of a bottom visible layer and a top two hidden layers, with connections only between adjacent layer units, but between the visible layer and the hidden layer is none connect to the connection instead of the directed connection.

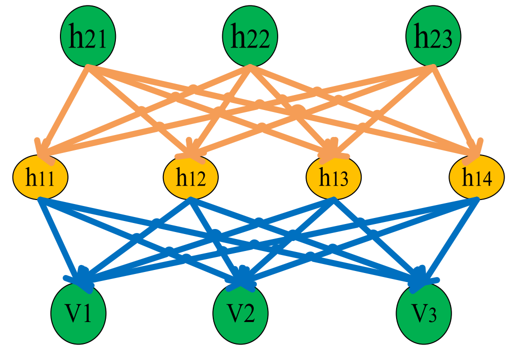

FIGURE 7. Deep Boltzmann machine model.

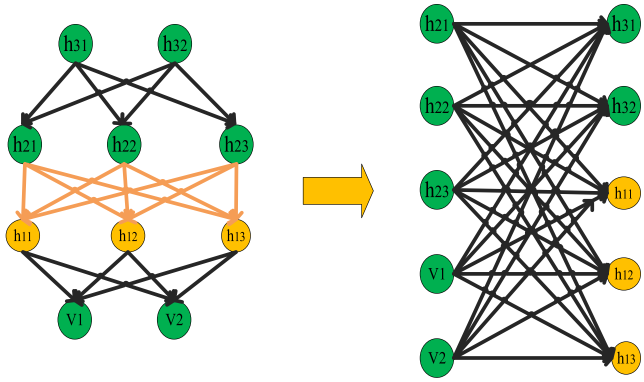

FIGURE 8. Rearrangement into a Deep Boltzmann Machine with a bipartite graph structure.

In the case where a deep Boltzmann machine contains a visible layer v and three hidden layers h1, h2 and h3, the joint probability is given by:

$$P(v, h^{(1)}, h^{(2)}, h^{(3)}) = e^{v,h^{(1)},h^{(2)},h^{(3)};\theta}/Z(\theta) \tag{12}$$

$$E(v, h^{(1)}, h^{(2)}, h^{(3)}; \theta) = -v^T W^{(1)}h^{(1)} - h^{(1)^T} W^{(2)}h^{(2)} - h^{(2)^T} W^{(3)}h^{(3)} \tag{13}$$

Compared to the energy function of RBM, the DBM energy function represents the connection between hidden units in the form of a weight matrix. DBM and RBM belong to the Boltzmann machine. There are some common advantages. As detailed in Figure 8, DBM can be reorganized into a bipartite graph with odd and even layers, with odd and even layers on either side. And the variables in the odd layer are independent of the conditions in the even layer.

DBM's bipartite graph structure description: The same formula for RBM conditional distribution can be used to determine the conditional distribution in DBM. Given the adjacent layer values, the elements within the layer are conditionally independent of each other. At the same time, the bipartite graph structure also makes Gibbs sampling more efficient in DBM.

C. CONVOLUTIONAL NEURAL NETWORKS

CNN is a feedforward neural network. For CNN, the medical image can be used as the original input of the bottom layer

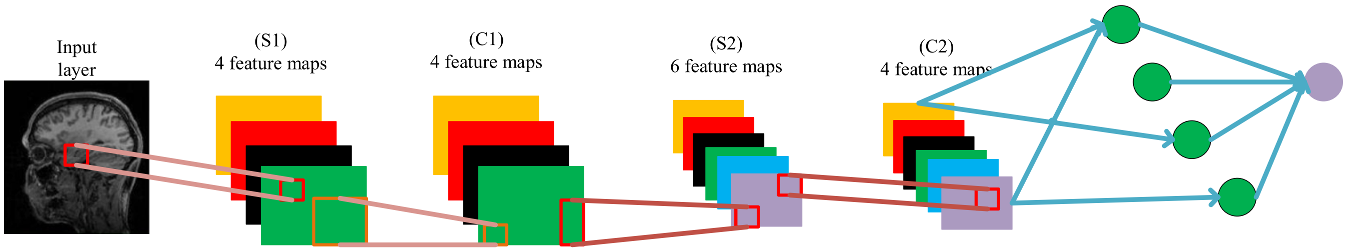

FIGURE 9. Example of convolution neural network.

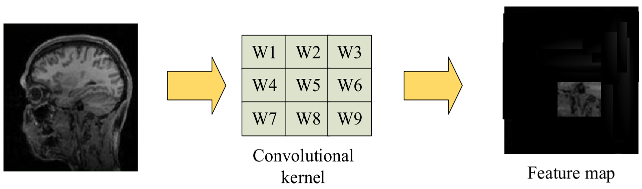

FIGURE 10. Convolutional layer.

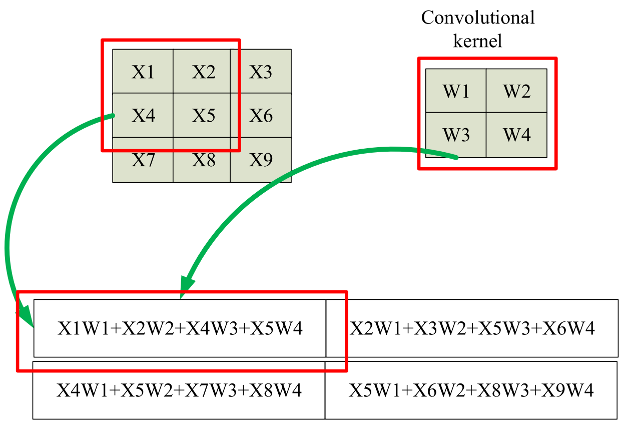

FIGURE 11. Two-dimensional convolution calculation.

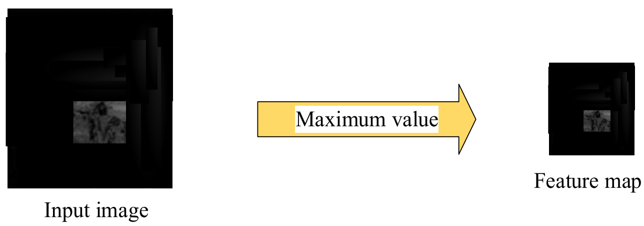

FIGURE 12. The pooling layer.

of the network, and then transmitted to the next layer in turn. Each layer extracts the most significant feature of the image data through a convolution kernel, and the output result is the classification and recognition result of the image. This multi-layered learning model can help discover the complex structure of data and improve the accuracy of image recognition.

The CNN is mainly composed of a convolution layer, a pooling layer, and a classifier, and the pooling layer may also be referred to as a down sampling layer. The convolutional layer and the pooled layer are considered to be two-dimensional layers, and the classifier is considered to be a one-dimensional layer. In CNN, each two-dimensional layer contains multiple planes, each plane consisting of neurons arranged in a two-dimensional array, and the output of the plane is called a feature map. The convolutional layer and the pooled layer alternate and the nonlinear mapping are adopted between the network layers. The convolutional layer to the pooled layer is a down sampling process, and the pooling layer to the convolutional layer is a convolutional filtering process. A convolution kernel extracts an image feature, so a feature map is obtained. The general convolutional neural network structure is shown in Figure 9. It can be seen from the Figure that this is a multi-layer network structure, and is composed of multiple parts in series, mainly including a convolutional layer, a pooled sampling layer, and a fully connected layer; whether it is a convolutional layer or a pooled sampling layer, they are all composed of multiple feature maps.

As shown in Figure 10, the convolutional layer convolves the target image through the convolution kernel to generate a feature map to achieve local feature perception and feature extraction [27]. Through training, each feature map obtained by the convolution operation is characterized by outstanding features in some aspects, such as edges, corners, line segments, and endpoints.

Convolution is a special kind of linear operation. The schematic diagram of two-dimensional convolution calculation is shown in Figure 11. Each element in the Figure represents the pixel value of the image.

In deep convolutional neural networks, the pooled sampling layer uses pooling to aggregate features at different locations, reducing the dimension or size of the feature map, and achieving translational invariance of the input image, improving feature robustness [28]. The common pooling methods are maximum pooling, average pooling, and random pooling. The scale of pooling is not too large, generally 22× or 33× pixels, which leads to the loss of excessive image details [29]. Figure 12 shows the pooling of the largest four pools of adjacent pixels in the image data.

After the convolutional layer or pooled sampling layer of the deep convolutional neural network, there is typically one or more fully connected layers before the output layer. Since the previous convolutional layer and the pooled sampling layer have already processed the image layer by layer, the size

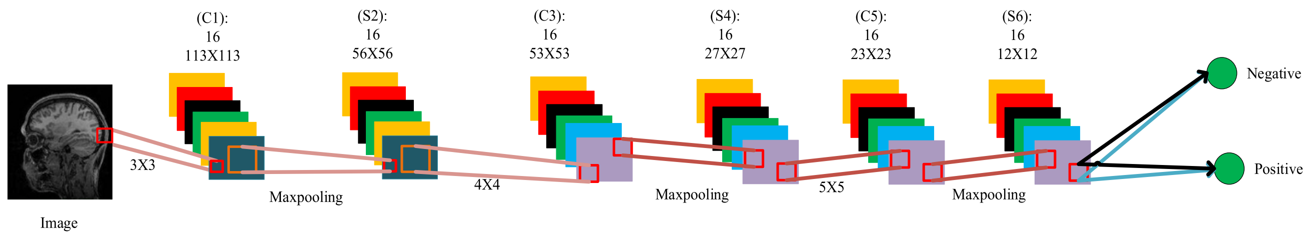

FIGURE 13. The structure of the adaptive convolution neural network.

of the feature map is within an acceptable range. In this case, using the fully connected layer does not bring a very large calculation. The quantity, the one or more consecutive fully connected layers form a shallow multi-layer perceptron, which serves as a classification.

III. ADAPTIVE CONVOLUTIONAL NEURAL NETWORK

A. ADAPTIVE CONVOLUTIONAL NEURAL NETWORK MODEL

The structure of the adaptive convolutional neural network model proposed in this paper is shown in Figure 13. The model consists of three convolutional layers (C1, C3, C5) and three pooling layers (S2, S4, S6) and consisting of a fully connected layer.

The input to the network is a brain image containing R channel information, G channel information, and B channel information for each image block.

The number of categories of the output layer is two, that is, lesions and non-lesions of the brain image.

Input layer: The input is a sample brain image of 227×227 pixels.

Convolution layer C1: There are 16 convolution kernels with a size of 3×3, the step size is 2, and a convolution kernel will get a feature map. Therefore, this layer consists of 16 feature maps, each feature map. The size of the graph is 113.

Pooling layer S2: S2 adopts the maximum pooling strategy, which is obtained after the C1 layer is down sampled. The size of the pooled area in S2 is 3×3, and the step size is 2. This layer consists of 16 feature maps with a size of 56×56.

Convolutional layer C3: There are 16 4×4 convolution kernels with a step size of 1. After convolution, 16 feature maps with a size of 53×53 will be obtained.

Pooling layer S4: S4 adopts the maximum pooling strategy, which is obtained after the C3 layer is down sampled. The pooled area has a size of 3×3 and a step size of 2, and consists of 16 feature maps with a size of 27×27.

Convolutional layer C5: It has 16 5×5 convolution kernels with a step size of 1, and consists of 16 feature maps with a size of 23×23.

Pooling layer S6: S6 adopts the maximum pooling strategy, which is obtained after the C5 layer is down sampled. The pooled area has a size of 3×3 and a step size of 2, and consists of 16 feature maps with a size of 12×12.

Output layer: This layer is a fully connected layer containing 2 neurons, each of which corresponds to a brain image of lesions and non-lesions: Benign is non-lesion and Malignant is lesion.

B. LOCAL CONNECTION AND PARAMETER SHARING CORE

The most important feature of convolutional neural networks is local connectivity and parameter sharing. The purpose is to reduce the number of CNN parameters, save training overhead, and speed up network training.

Traditional neural networks are fully connected neural networks, that is, all neurons in two adjacent layers are connected, and matrix multiplication is used to establish the connection between input and output. There is an interaction between each input unit and the output unit. However, unlike fully connected neural networks, CNN takes a local connection to interact.

The so-called local connection means that each layer of neurons is only connected to a part of the neuron area of the upper layer, and this part of the response area is also called a local receptive field. The local connection properties of CNN result in a significant reduction in inter-layer parameters and better study of local features of the image. For example, if a network has m inputs and n outputs, then for a fully connected network, $m \times n$ parameters are required and the time complexity is $O(m \times n)$ x. For CNN, to limit the number of connections each output has to k, then the local connection method only requires $k \times n$ parameters and the running time of $O(k \times n)$.

Looking at each of the graphs of Figure 14 from the bottom up, taking the input unit x3 as an example, the local connection and the full connection of the neural network are described by analyzing the affected output unit in s. Figure 14(a) shows a partial connection. As can be seen from (a), when s is generated by convolution with a kernel width of 3, only three output units s1, s2, s3 are affected by x. And in (b), s is generated by matrix multiplication, taking a full-join operation, no longer a local connection, so all outputs are affected by x3.

Parameter sharing refers to the use of the same weighting parameters in multiple local receptive fields of a convolutional neural network model. Parameter sharing ensures that CNN only needs to learn a set of parameters, rather than requiring a separate set of parameters for each location.

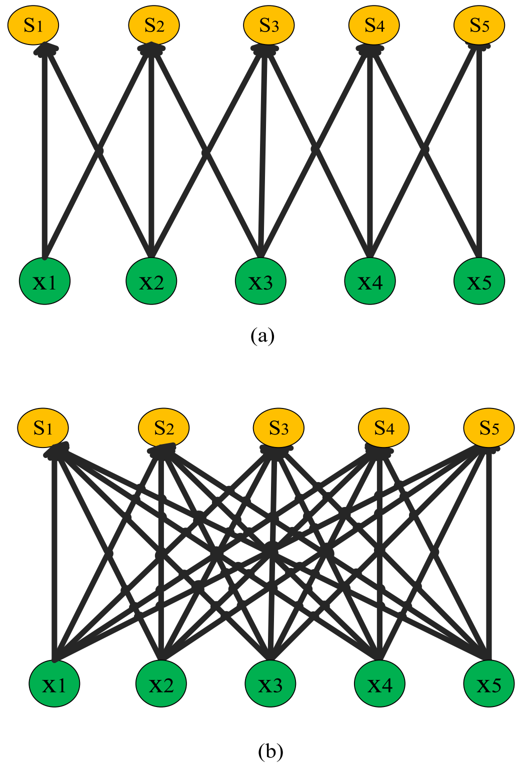

FIGURE 14. Local connected and fully connected.

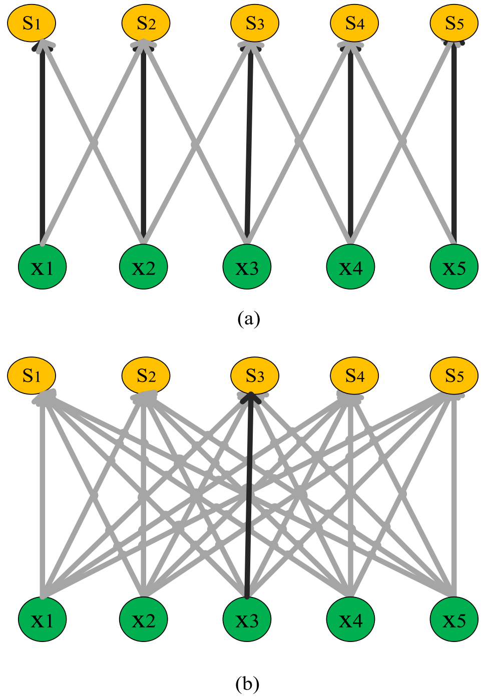

FIGURE 15. Parameter sharing.

The nature of parameter sharing makes CNN translationally invariant. Figure 15 depicts the parameter sharing of CNN. In Figure 15(a), the black arrow indicates that the same parameter is used at all input positions, that is, parameter sharing. In Figure 15(b), the black arrow indicates the use of the intermediate elements of the weight matrix in the fully connected model. Parameter sharing is not used in the fully connected model, so the parameters are used only once.

C. OPTIMIZATION METHOD OF CONVOLUTIONAL NEURAL NETWORKS

Batch Normalization algorithm is one of the most exciting innovations in optimizing deep convolutional neural networks. It is a data normalization algorithm proposed by literature [30] to alleviate the covariance migration phenomenon.

The BN algorithm is essentially a learnable normalization method that uses a normalization step to have each layer have input data that is subject to the same distribution. The advantages of the BN algorithm are: First, it reduces the dependence of the gradient on the parameter scale or initial value, so that it can be trained with a large learning rate to speed up the network convergence; secondly, the BN also normalizes the CNN model, reducing Dropout. Finally, it makes it possible to use a saturated nonlinear activation function in the saturation model.

The BN algorithm introduces two learnable parameters for each neuron and performs transform reconstruction. The formula is as follows:

(1) Normalization:

$$\hat{x}^{(k)} = (x^{(k)} - E(x^{(k)}))/\sqrt{Var(x^{(k)})} \tag{14}$$

In the training process, the random gradient descent method is adopted on the batch training data. The $E(x^{(k)})$ in the formula (14) refers to the average of each training data neuron $x^{(k)}$; the $\sqrt{Var(x^{(k)})}$ represents the standard deviation of the activation degree of each batch of data neurons.

(2) Transform reconstruction:

$$y^{(k)} = \gamma^{(k)}\hat{x}^{(k)} + \beta^{(k)} \tag{15}$$

$$\gamma^{(k)} = \sqrt{Var(x^{(k)})} \tag{16}$$

$$\beta^{(k)} = E(x^{(k)}) \tag{17}$$

Each neuron has a pair of such parameters γ and β. γ and β need to be trained by BP algorithm.

(3) The forward propagation process of the BN network layer is:

$$\begin{aligned}\mu_B &\leftarrow (1/m)\sum x_i \\ \sigma_B^2 &\leftarrow (1/m)\sum (x_i - \mu_B)^2 \\ \hat{x}_i &\leftarrow (x_i - \mu_B)/(\sqrt{\sigma_B^2} + \varepsilon) \\ y_i &\leftarrow \gamma\hat{x}_i + \beta = BN_{\gamma,\beta}(x_i)\end{aligned} \tag{18}$$

The input value is $x_1, \ldots, x_n$, where m is the number of batches; μ_{B} is the mean, σ_B^2 is the variance; $y_i = BN_{\gamma,\beta}(x_i)$ is the output.

For the case where the ReLU neural unit may die during training and the output has an offset phenomenon [31], a Parametric Rectified Linear Unit is used instead of ReLU as a nonlinear activation function of the neuron. PReLU can

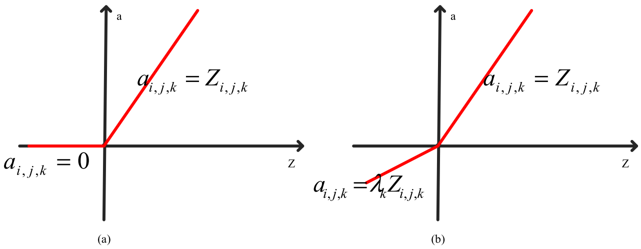

FIGURE 16. **Comparison of ReLU and PReLU functions.**

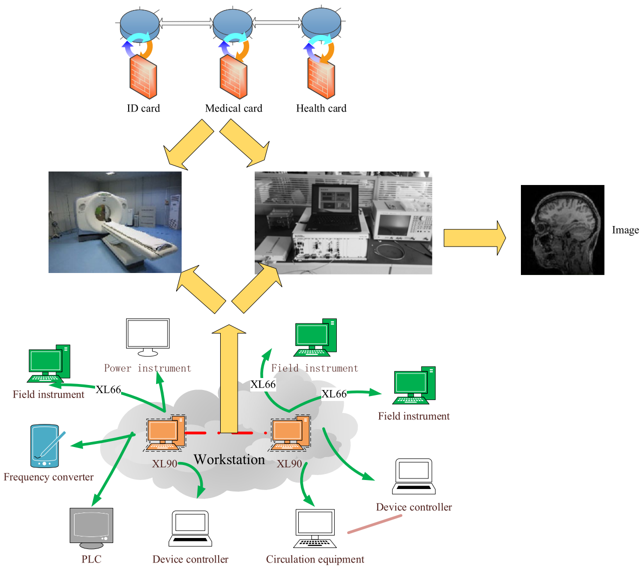

FIGURE 17. **The designing diagram of an acquisition device based on the Internet of Things.**

adaptively learn parameters from data, and has the characteristics of low error rate and fast convergence speed, which can improve the recognition accuracy when the extra calculation cost is negligible.

PReLU introduces only a very small number of additional parameters, the number of which is the same as the number of channels in the entire network, so there is no additional risk of overfitting. In addition, PReLU can also train with other parameters simultaneously through backpropagation.

IV. EXPERIMENTS AND RESULTS

A. DATABASE DESCRIPTION

With the rapid development of IoT-related industries and the widespread application of intelligent mobile terminals, mobile terminals are playing an increasingly important role in the rich Internet of Things applications. The premise of IoT application is the collection of massive information, and the combination of multi-source information collection and intelligent mobile devices allows the Internet of Things to be further integrated into people's production and life. In this paper, IoT technology is used to design a collection device based on Internet of Things technology, and medical images are collected as a research database. Figure 17 is a design diagram of an acquisition device based on the Internet of Things.

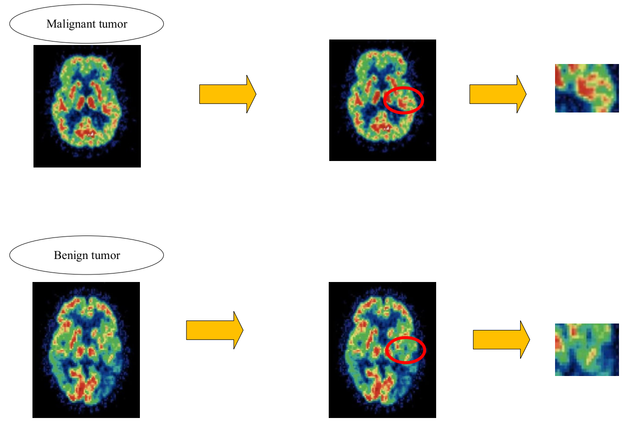

FIGURE 18. **The diagram of breast mammography with preprocessing.**

The following is the process of collection:

(1) Verify the identity of the patient and verify by entering (2).

(2) Let the patient put on protective clothing into our collection device.

(3) The collecting personnel operate the collecting program and collect multiple sets of images.

B. IMAGE PREPROCESSING

In order to improve the performance of the convolutional neural network, preprocessing is a necessary step in building the data set. In this paper, the lesion areas are extracted as fixed-size regions of interest and then localized using global contrast normalization.

Figure 18 illustrates the pre-processing steps for extracting and normalizing the ROI. As shown in Figure 18, the first and second rows give the lesion and non-lesion brain images, respectively: where (a) represents the original X-ray image; (b) represents the lesion location and boundary as indicated by the expert annotation; (c) indicates the ROI after cropping.

The region of interest refers to the area in the image that needs to be processed in a box or the like. In this paper, the region of interest is also the lesion area. Therefore, in order to simplify the training process, the ROI can be automatically positioned and cropped to 227 $\times$ 227 pixels. The region was used as the input of the convolutional neural network model, in which 634 non-lesion brain images and 599 lesion brain images were selected, the data set was divided into training set and test set, and the recognition effect of CNN model was evaluated by five-fold cross-validation.

The best way to generalize the convolutional neural network model is to use more data for training. Training the CNN model in this case will make the network converge slowly and the recognition effect is relatively poor. The solution to this problem is data enhancement, which helps to build a simpler and better-promoted robust model [32]. In order to achieve data enhancement, a common method is to add noise or geometrically transform an existing image. This paper mainly uses the geometric transformation method for data enhancement. In this experiment, the training data is

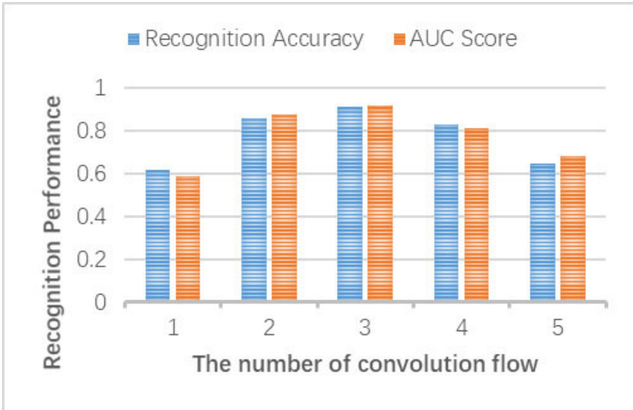

FIGURE 19. Recognition accuracy rate and AUC score with the number of convolution flow curve.

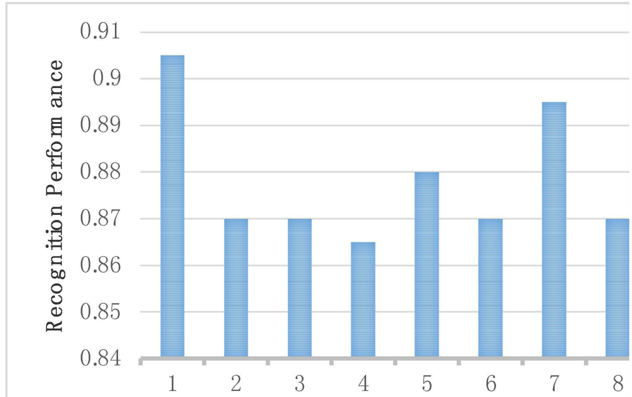

FIGURE 20. The recognition accuracy of nine different configurations.

expanded to 4932 by randomly performing translation, rotation and flip operations on the brain image in the training set: where the random translation is in the range [−6, 6]; the rotation is through the two-dimensional Rotate [90°, 180°, 270°]. Such an enhancement of data not only makes it easier for CNN to extract brain image features, but also maintains brain image features.

C. SELECTION OF EXPERIMENTAL PARAMETERS

As a multi-layer perceptron, the convolutional neural network is an important factor in determining the efficiency of network classification. Too many network layers can cause problems such as increased training parameters, slow training speed, and long training time. It is also prone to over-fitting. However, if the number of network layers is too small, the network cannot discover the complex structure of the data and extract accurate data features, which will make the network classification accuracy rate low. Therefore, the number of different network layers has an important impact on network identification efficiency.

When this paper selects experimental parameters, trains the model for each experimental parameter change, and have achieved the optimal goal.

In order to select a suitable number of network layers, a form of increasing the convolution flow is used, and a convolution flow includes a convolution layer and a pooling layer. Based on the stochastic gradient descent method, experiments are started from one convolution stream. The experimental results are shown in Figure 19. As can be seen from Figure 19, as the number of convolutional streams increases, the recognition accuracy and AUC values will gradually increase and then decline. When the number of convolution streams is 3, the recognition is best. After selecting the number of convolutional streams, select one fully connected layer for classification, and the number of neurons in the fully connected layer is 2.

To verify the impact of the number of convolution kernels on network performance, assume that each convolutional layer has n convolution kernels, the convolutional layer location is i, and the number of convolutional layers in the first layer is fixed at 16, where n={64, 32,16}, i={3,2}, there are 9 different configurations, namely {16,16,16},{16,16,32}, {16,16,64},{16, 32, 16}, {16, 32, 32}, {16, 32, 64}, {16, 64, 16}, {16, 64, 32}, {16, 64, 64}. The network model is trained to iterate 5000 times, and the trained model is saved every 1000 iterations. The initial learning rate is set to 1.0×10-3, and the learning rate attenuation mode is Poly, power=0.5.

The experimental results in Figure 20 show that the recognition accuracy of the nine different configurations is small. When the number of convolution kernels in the three convolutional layers is {16, 16, 16}, the performance of the network is optimal, and the recognition accuracy reaches 0.905.

D. OPTIMIZATION OF NETWORK MODEL SELECTION

In this chapter, the volume normalization algorithm and the parameterized linear correction unit PReLU are used to optimize the convolutional neural network, and four models are named:

(1) CNN+ReLU: The convolution layer activation function uses ReLU;

(2) CNN+PReLU: The convolutional layer activation function adopts PReLU;

(3) CNN+BN+ReLU: After the convolution operation in the convolutional layer, the BN algorithm is used for batch normalization, and then the nonlinear function is output through the activation function ReLU.

(4) CNN+BN+PReLU: After the convolution operation in the convolutional layer, the BN algorithm is used for batch normalization, and then the PReLU is used for nonlinear output.

For each optimization algorithm, this paper trains the model. As shown in Table 2, the above four models were evaluated in the training time, recognition accuracy, sensitivity, specificity and AUC of the data set experiments. The sensitivity and specificity respectively reflected the misdiagnosis rate of CNN network and missed diagnosis rate.

The experimental results in Table 2 show that under the same training conditions, the CNN+BN+PReLU model

TABLE 2. Comparison of experimental results of different CNN models.

Model	CNN+ Relu	CNN+ PRelu	CNN+ BN+Relu	CNN+ BN+PRelu
Time	619.311	523.921	181.321	251.012
Accuracy	0.825	0.855	0.836	0.912
Sensitivity	0.861	0.912	0.945	0.963
Specificity	0.791	0.809	0.732	0.875
AUC	0.831	0.867	0.853	0.918

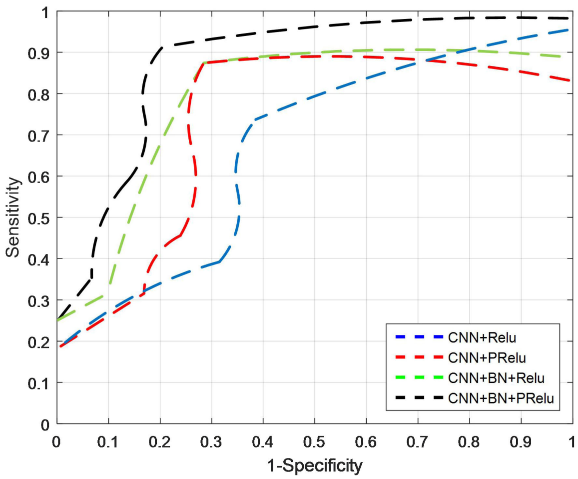

FIGURE 21. Comparison of ROC curves for different CNN models.

is superior to the other three models CNN+ReLU, CNN+PReLU and CNN in recognition accuracy, sensitivity, specificity and AUC score. CNN+BN+ReLU, the accuracy of the model is 8.5% higher than that of the CNN+ReLU model before optimization, achieving a better recognition effect. In the training time, because the CNN+BN+PReLU model is added to PReLU, the time consumption is slightly higher than that of CNN+BN+ReLU, but it is still about 71.83% lower than CNN+ReLU and about 68.25% less than CNN+PReLU.

The ROC curve visually reflects the classification performance of different CNN networks. AUC is the area under the curve of the ROC. The closer the AUC value is to 1, the better the classification effect. Figure 21 shows the ROC plots for the four models with 1-specificity on the abscissa and sensitivity on the ordinate.

It can be seen from Figure 21 that the CNN+BN+PReLU model is closest to the point (0, 1), that is, the recognition effect is the best. It can also be seen that although the optimized network models CNN+PReLU and CNN+BN+ReLU have been significantly improved in each indicator, there is still no improved CNN+BN+PReLU model and high recognition performance.

The BN algorithm is applied as a network layer in CNN. The recognition performance of the BN layer before and after the activation function is different. In order to select the optimal adaptive network structure, as shown in Table 3, this part will conduct experimental comparisons on this problem.

TABLE 3. Comparison of recognition effect of BN algorithm before and after PReLU.

Model	CNN+ PRelu+BN	CNN+BN+PRelu
Time	251.321	251.012
Accuracy	0.865	0.912
Sensitivity	0.913	0.963
Specificity	0.832	0.875
AUC	0.873	0.918

The convolutional neural network model that performs non-linear output by the activation function PReLU after the convolution operation and then performs batch normalization by the BN algorithm is called the CNN+PReLU+BN model.

The experimental results in Table 3 show that the training time of the two models is basically the same, but the accuracy, sensitivity and AUC value of CNN+BN+PReLU are higher than the former by 3.5%, 5%, 5% and 0.03, respectively. Compared with the training parameters, the performance of BN before PReLU is better. Therefore, CNN+BN+PReLU model is chosen as the adaptive convolutional neural network model.

E. PERFORMANCE COMPARISON OF NETWORK MODELS

In order to further verify the superiority of the CNN+BN+PReLU method in this paper, this paper counts the brain image classification results of the following methods:

(1) The literature [18] proposes a method based on the feature extraction of gray level co-occurrence matrix, and extracts nine eigenvalues of brain images respectively, and then classifies them through BP neural network.

(2) Literature [33] extracts brain image features based on deep convolutional neural networks and classifies them.

(3) The literature [34] proposed a method based on high-dimensional multi-resolution histogram feature extraction, which extracted 192 feature values of brain images and then classified them by SVM.

(4) The literature [35] extracts 56 feature values of the shape and texture of the brain image respectively, and then classifies them by SVM.

(5) The literature [36] extracts 23 eigenvalues such as morphological features and spatial positions of brain images, and then uses genetic algorithm and BP neural network classification to classify.

This paper compare the performance of the adaptive convolutional neural network model CNN+BN+PReLU with the two classical convolutional neural network models LeNet-5 and AlexNet and the methods in the above literature.

The traditional manual extraction of single features has a poor classification effect. In order to obtain better classification results, it is necessary to extract features in many aspects, but the artificial feature extraction is more complicated and the generalization ability is poor. The classification method based on convolutional neural network it can be simpler, more convenient, and has a strong self-learning ability, but the

TABLE 4. Performance comparison between proposed method and others.

Model	Accuracy	Sensitivity	Specificity	AUC
LeNet-5[37]	0.745	0.752	0.751	0.731
AlexNet[38]	0.605	0.610	0.602	0.582
MRH+SVM[34]	0.722	0.732	0.715	0.725
GLCM+BP[36]	0.861	0.825	0.861	0.857
Texture+SVM[35]	0.891	0.897	0.856	0.885
GA+BP[18]	0.911	0.931	0.925	0.905
CNN[33]	0.721	0.732	0.787	0.712
CNN+BN+PRelu	0.935	0.951	0.921	0.926

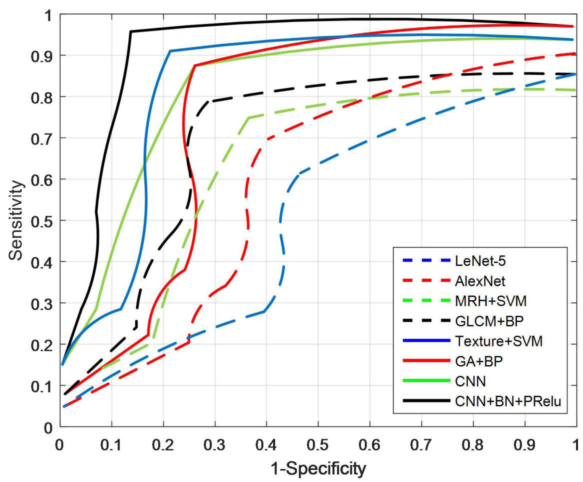

FIGURE 22. Comparison of ROC curve with classical CNNs.

accuracy needs to be further improved. It can be seen from Table 4 and Figure 22 that compared with the two classical convolutional neural network models and the above references, the model has higher superiority in accuracy, sensitivity, specificity and AUC value, and can be effective to improve the recognition accuracy and reduce the misdiagnosis rate and missed diagnosis rate of brain image lesions.

V. CONCLUSION

In today's big data background, brain image recognition has become an important research direction in the field of computer vision. Different from the early machine learning methods, convolutional neural network can not only extract deep features of images, but also combine feature extraction with classification process to improve recognition efficiency. This paper first summarizes the research background, significance and current situation of this topic, and then discusses the inadequacies of convolutional neural network. Secondly, it introduces the theory of brain image recognition and convolutional neural network. Finally, it focuses on the structure and local connection of convolutional neural networks, share ideas with parameters, parameter update process, and problems. Based on the study of convolutional neural networks, this paper proposes an adaptive convolutional neural network model. Aiming at the problems of slow convergence and low accuracy during training, this paper proposes to use the batch normalization algorithm and the parametric linear correction unit PReLU to improve it, and prove that the combination of the two can effectively shorten the training time and improve the network. The generalization performance shortens the network training time and significantly improves the recognition accuracy of brain images. The results show that the method not only has a significant improvement in recognition accuracy, but also reduces the rate of misdiagnosis and missed diagnosis.

REFERENCES

[1] M. Kallenberg, K. Petersen, M. Nielsen, A. Y. Ng, P. Diao, C. Igel, C. M. Vachon, K. Holland, R. R. Winkel, N. Karssemeijer, and M. Lillholm, "Unsupervised deep learning applied to breast density segmentation and mammographic risk scoring," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1322–1331, May 2016.

[2] X. Sun, L. Xiao-Guang, L. Jia-Feng, and Z. Li, "Review on deep learning based image super-resolution restoration algorithms," *Acta Automatica Sinica*, vol. 43, no. 5, pp. 697–709, 2017.

[3] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, pp. 60–88, Dec. 2017.

[4] D. Shen, G. Wu, and H. Suk, "Deep learning in medical image analysis," *Annu. Rev. Biomed. Eng.*, vol. 19, no. 1, pp. 221–248, Jun. 2017.

[5] A. Bourouis, M. Feham, M. Hossain, and L. Zhang, "An intelligent mobile based decision support system for retinal disease diagnosis," *Decis. Support Syst.*, vol. 59, no. 2, pp. 341–350, Mar. 2014.

[6] S. B. Jiang, P. K. Wong, R. Guan, Y. Liang, and J. Li, "An efficient fault diagnostic method for three-phase induction motors based on incremental broad learning and non-negative matrix factorization," *IEEE Access*, vol. 7, pp. 17780–17790, 2019.

[7] R. Chaves, J. Ramírez, J. M. Górriz, and I. A. Illán, "Functional brain image classification using association rules defined over discriminant regions," *Pattern Recognit. Lett.*, vol. 33, no. 12, pp. 1666–1672, 2012.

[8] J. Wang, X. Feng, and T. Yu, "A geometric approach to support vector regression and its application to fermentation process fast modeling," *Chin. J. Chem. Eng.*, vol. 20, no. 4, pp. 715–722, 2012.

[9] D. Avci, M. K. Leblebicioglu, M. Poyraz, and E. Dogantekin, "A new method based on adaptive discrete wavelet entropy energy and neural network classifier (ADWEENN) for recognition of urine cells from microscopic images independent of rotation and scaling," *J. Med. Syst.*, vol. 38, no. 2, pp. 7–15, 2014.

[10] M. Peker, "A decision support system to improve medical diagnosis using a combination of k-medoids clustering based attribute weighting and SVM," *J. Med. Syst.*, vol. 40, no. 5, p. 116, 2016.

[11] I. B. de Medeiros, M. A. S. Machado, W. J. Damasceno, A. M. Caldeira, R. C. dos Santos, and J. B. da Silva Filho, "A fuzzy inference system to support medical diagnosis in real time," *Procedia Comput. Sci.*, vol. 122, pp. 167–173, Jan. 2017.

[12] M. S. Haghshenas, M. J. Ashraf, and B. Khademi, "Chemokine and chemokine receptor patterns in patients with benign and malignant salivary gland tumors: A distinct role for CCR7," *Eur. Cytokine Netw.*, vol. 28, no. 1, pp. 27–35, Jan. 2017.

[13] P. Zheng, Y. Wang, X. Xu, and S. Q. Xie, "A weighted rough set based fuzzy axiomatic design approach for the selection of AM processes," *Int. J. Adv. Manuf. Technol.*, vol. 91, nos. 5–8, pp. 1977–1990, 2017.

[14] P. Tian, W. Zhang, H. Zhao, Y. Lei, L. Cui, W. Wang, Q. Li, Q. Zhu, Y. Zhang, and Z. Xu, "Intraoperative diagnosis of benign and malignant breast tissues by Fourier transform infrared spectroscopy and support vector machine classification," *Int. J. Clin. Exp. Med.*, vol. 8, no. 1, pp. 972–981, 2015.

[15] Q. Su, Y. Niu, Q. Wang, and G. Sheng, "A blind color image watermarking based on DC component in the spatial domain," *Optik-Int. J. Light Electron Opt.*, vol. 124, no. 23, pp. 6255–6260, Dec. 2013.

[16] S. Hao, L. Yang, and Y. Shi, "Data-driven car-following model based on rough set theory," *IET Intell. Transp. Syst.*, vol. 12, no. 1, pp. 49–57, Feb. 2018.

[17] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, 1996.

[18] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[19] D. Yu and L. Deng, "Deep learning and its applications to signal and information processing [exploratory DSP]," *IEEE Signal Process. Mag.*, vol. 28, no. 1, pp. 145–154, Jan. 2011.

[20] H. Li, "Deep learning for natural language processing: Advantages and challenges," *Nat. Sci. Rev.*, vol. 5, no. 1, pp. 24–26, 2018.

[21] S. Pouyanfar and S.-C. Chen, "Automatic video event detection for imbalance data using enhanced ensemble deep learning," *Int. J. Semantic Comput.*, vol. 11, no. 1, pp. 85–109, 2017.

[22] H. Venkateswara, S. Chakraborty, and S. Panchanathan, "Deep-learning systems for domain adaptation in computer vision: Learning transferable feature representations," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 117–129, Nov. 2017.

[23] X. Gao, F. Li, and K. Wan, "Accelerated learning for restricted Boltzmann machine with a novel momentum algorithm," *Chin. J. Electron.*, vol. 27, no. 3, pp. 483–487, May 2018.

[24] W. Pannakkong, S. Sriboonchitta, and V.-N. Huynh, "An ensemble model of Arima and ann with restricted Boltzmann machine based on decomposition of discrete wavelet transform for time series forecasting," *J. Syst. Sci. Syst. Eng.*, vol. 27, no. 5, pp. 690–708, 2018.

[25] A. M. Sheri, A. Rafique, W. Pedrycz, and M. Jeon, "Contrastive divergence for memristor-based restricted boltzmann machine," *Eng. Appl. Artif. Intell.*, vol. 37, pp. 336–342, Jan. 2015.

[26] R. Salakhutdinov and G. Hinton, "Deep Boltzmann machines," *J. Mach. Learn. Res.*, vol. 5, no. 2, pp. 1967–2006, 2009.

[27] J.-H. Lee, D.-H. Kim, S.-N. Jeong, and S.-H. Choi, "Diagnosis and prediction of periodontally compromised teeth using a deep learning-based convolutional neural network algorithm," *J. Periodontal Implant Sci.*, vol. 48, no. 2, pp. 114–123, 2018.

[28] C. Henry, S. M. Azimi, and N. Merkle, "Road segmentation in SAR satellite images with deep fully convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 12, pp. 1867–1871, Dec. 2018.

[29] Z. Song, Y. Liu, R. Song, Z. Chen, J. Yang, C. Zhang, and Q. Jiang, "A sparsity-based stochastic pooling mechanism for deep convolutional neural networks," *Neural Netw.*, vol. 105, pp. 340–345, Sep. 2018.

[30] D.-X. Xia, S.-Z. Su, L.-C. Geng, G.-X. Wu, and S.-Z. Li, "Learning rich features from objectness estimation for human lying-pose detection," *Multimedia Syst.*, vol. 23, no. 4, pp. 515–526, 2017.

[31] Y.-D. Zhang, C. Pan, X. Chen, and F. Wang, "Abnormal breast identification by nine-layer convolutional neural network with parametric rectified linear unit and rank-based stochastic pooling," *J. Comput. Sci.*, vol. 27, pp. 57–68, Jul. 2018.

[32] Z. Wu, Y. Huang, L. Wang, X. Wang, and T. Tan, "A comprehensive study on cross-view gait based human identification with deep CNNs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 2, pp. 209–226, Feb. 2016.

[33] B. Shi, X. Bai, and C. Yao, "Script identification in the wild via discriminative convolutional neural network," *Pattern Recognit.*, vol. 52, pp. 448–458, Apr. 2016.

[34] J. Wang, X. Li, Y. Zhang, and X. Zhang, "Adaptive decomposition method for multi-modal medical image fusion," *IET Image Process.*, vol. 12, no. 8, pp. 1403–1412, Aug. 2018.

[35] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, no. 7, pp. 257–269, 2010.

[36] Z. Lian, X. Jing, X. Wang, H. Huang, Y. Tan, and Y. Cui, "DropConnect regularization method with sparsity constraint for neural networks," *Chin. J. Electron.*, vol. 25, no. 1, pp. 152–158, Jan. 2016.

[37] R. Anil, K. Manjusha, S. S. Kumar, and K. P. Soman, "Convolutional neural networks for the recognition of Malayalam characters," *Int. J. Comput. Sci. Math.*, vol. 5, no. 2, pp. 209–218, 2015.

[38] M. Long, C. Ouyang, H. Liu, and Q. Fu, "Image recognition of Camellia oleifera diseases based on convolutional neural network & transfer learning," *Trans. Chin. Soc. Agricult. Eng.*, vol. 34, no. 18, pp. 194–201, 2018.

YUXI LIU was born in Heilongjiang, China, in 1992. He received the bachelor's degree from Harbin Normal College, in 2015. He is currently pursuing the master's degree with the University of Tasmania, Australia. His research interests include artificial intelligence image recognition and big data.

JUN XIONG was born in Nanchang, Jiangxi, China, in 1989. He received the bachelor's degree from the Jiangxi University of Finance and Economics College, in 2011. He is currently pursuing the master's degree with The University of Sydney in Australia. His research interests include artificial intelligence image recognition, machine learning pricing prediction, visualization, and analytics.